Fall 12-1-2003

Data Warehouse Design and Implementation for FedEx Growth & Ranking Reports

Ranjith Boggarapu
Dakota State University

Data Warehouse Design and Implementation for FedEx Growth & Ranking Reports

By

Ranjith Boggarapu

A project submitted in partial fulfillment of the requirements for the

Master of Science in Information Systems

Dakota State University

2003

MSIS
PROJECT APPROVAL FORM

Student Name: Ranjith Boggarapu

Expected Graduation Date: December 2003

Master's Project Title: Data Warehouse Design and Implementation for FexEx Growth & Ranking Reports

Date Project Plan Approved: _____

Date Project Coordinator Notified and Grade Submitted: _____

Approvals/Signatures:

Student: B. Ranjith Date: 01/02/2004

Faculty supervisor: Date: 1/6/04

Committee member: Date: 1/6/04

Committee member: Date: 1/6/04

Copies to:
Original Attached to Written Report
Copies to: Advisor, Graduate Coordinator, and Student

Abstract

The objective of this project is to design and implement a database for a data warehouse meeting the business user's need and requirements for generating a report. This report handles the project right from the requirements gathering, analyzing and developing a data model to support the report requirements.

The project is implemented in tiered approach by summarizing the source tables by every tier and at the same time getting all the required data in to a consistent format. During this initial phase of rollout, the data model will support the Growth & Ranking requirements and is designed to be very flexible to fit the future enhancements and data requirements. This report discusses in detail about the technologies used to implement the database and its applicability in the current project.

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Project

Sales Planning & Analysis (SP&A) division of FedEx has divided its market into territories based on the geographical location and has assigned an employee to each territory. These territories will in turn roll up to the organizations. The territorial hierarchy is as below:

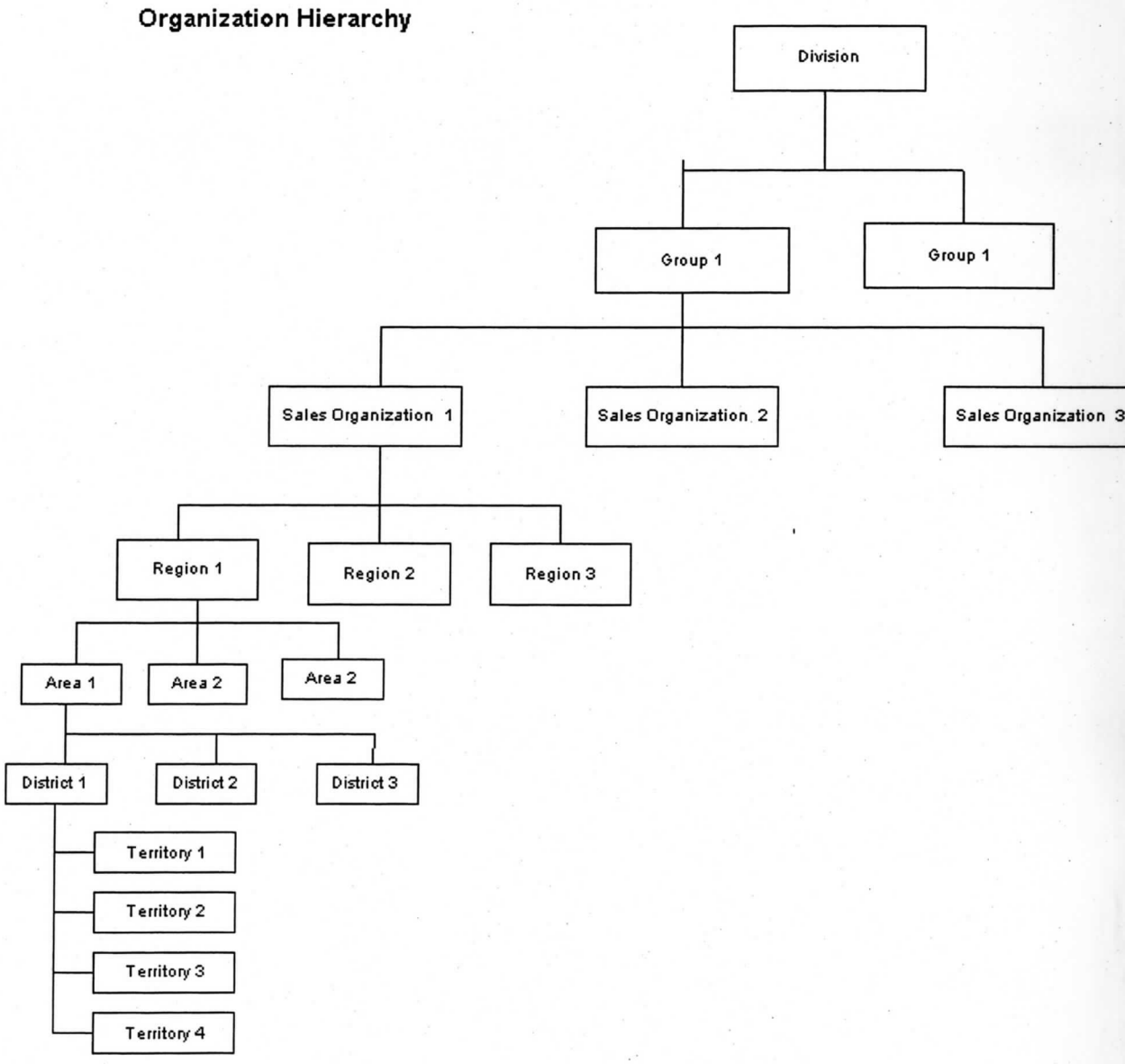


Figure 1-1 Organization Hierarchy

An employee is assigned at every level in the hierarchy except at area and shall be responsible for the Sales Organization/Region/ Area/District/Territory. The revenue generated in his/her assigned organization/region/district/territory is considered for their compensation. The sales representative is assigned at the lowest level of the organization, which is territory. He/she is held responsible for the accounts in their territory and shall meet the targets specified at the beginning of the year. Employees in the higher hierarchy are held responsible for the revenue generated by his/her subordinates.

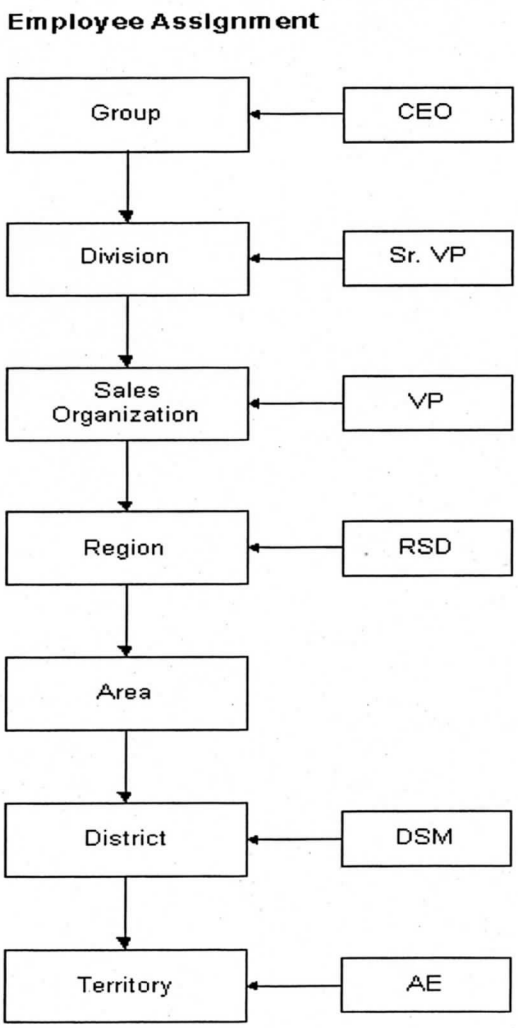


Figure 1-2 Employee Assignment

The target of a sales representative is measured in two metrics: Revenue & Volume. Revenue is the amount collected from the clients for shipping the packages during a certain period. Volume is the number of packages shipped during a certain period. A representative should meet both the revenue target and volume target.

Revenue generated by all the clients in a territory will be considered for calculating compensation of the sales representative. In the same way, revenue generated by clients of the entire district will be considered for the District Sales Manager's compensation. Compensation for employees in the hierarchy is computed in the same fashion.

The revenue for three service categories shall be reported to the users

- Express
- Ground
- International

The details of shipments and their revenue will be stored in various source systems and has to be integrated into one common repository.

Customers are assigned to a 9-digit unique account number throughout the organization. These 9-digit account numbers are grouped together and are assigned to a 5-digit account number. This 5-digit account number is otherwise called as a national account number and is a unique account number in the

national accounts. Ex: A customer like Dell will have numerous branches throughout the country and falls under different territories. Each Dell subdivision is assigned with a 9-digit account number, all these 9-digit account numbers are categorized, and a 5-digit account number is assigned to the Dell Corporation. This kind of categorization helps to find the revenue generated by the Dell in all over the country.

ISH (Information Super Hub) is a subdivision of FedEx Services and its major objective is to provide with any kind of information requested by the end user. Role of ISH is to serve as a data warehouse division for entire FedEx. ISH will be uploaded with the raw data on frequent interval via a batch process from all the sub-systems. Once the data is uploaded, ISH team integrates to serve as backbone.

This project is started as a step to automate two reports that are commonly generated by the sales professional on a day-to-day basis. The first report "Growth" gives the details of the revenue or volume in the territory / district / region / organization / division / group. This report help a sales professional/management to access the performance of the individual and see if he/she is meeting the targets set by the marketing division at the beginning of the year. The second report "Ranking" gives a list of top 10 customers who have generated highest revenue or volume during the selected period. At the same time, this report gives the bottom 10 customers who have generated lowest revenue or volume in the period select. This report helps a sales professional to emphasize

more on the customers who are not doing well with FedEx and gives an opportunity to provide some benefits to the top customers.

This report discusses requirements and analyses the data requirements to support report features. Once data requirements are analyzed, a Data Model is designed and the tables are created in Teradata Data warehouse in compliance with the Data Model. After creating tales, they are populated with the data. Due to the huge amount of data in the source tables, I have used the tiered approach where the data is summarized as data moves from one tier to other. The data in the data warehouse is de-normalized to an extent of the easier and faster access to data.

1.2 FedEx

Frederick W. Smith has developed a profound interest in his Arkansas Aviation Sales, located in Little Rock, Arkansas. While operating his new firm he has faced a tremendous difficulty in getting packaged and other airfreight delivered within one or two days. This problem has motivated him to do the necessary research for resolving the inefficient distribution system. Thus, the idea for Federal Express was born a company that revolutionized global business practices and now defines speed and reliability.

Federal Express was so-named due to the patriotic meaning associated with the word "Federal," which suggested an interest in nationwide economic activity. At that time, Smith hoped to obtain a contract with the Federal Reserve Bank and,

although the proposal was denied, he believed the name was a particularly good one for attracting public attention and maintaining name recognition.

Today FedEx is led by FedEx Corporation, which provides strategic direction and consolidated financial reporting for six independent operating companies that compete collectively under the FedEx name worldwide:

- FedEx Express
- FedEx Ground
- FedEx Freight
- FedEx Custom Critical
- FedEx Trade Networks
- FedEx Services

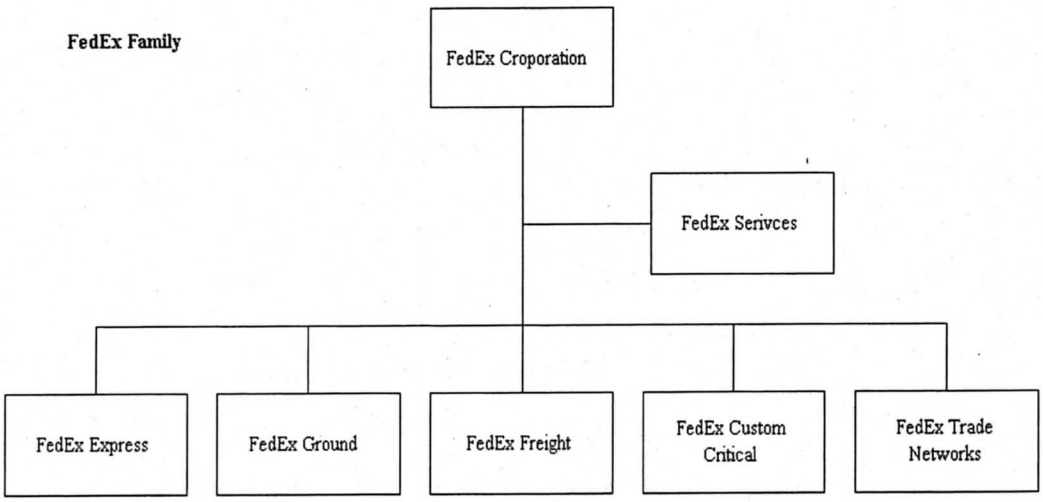


Figure 1-3 FedEx Family

1.2.1 FedEx Express

The company incorporated in June 1971 and officially began operations on April 17, 1973, with the launch of 14 small aircraft from Memphis

International Airport. On that night, Federal Express delivered 186 packages to 25 U.S. cities from Rochester, NY, to Miami, FL. Later the company headquarters was moved to Memphis, TN

FedEx Express is the world's largest express transportation company, providing fast and reliable delivery to every U.S. address and to more than 210 countries. FedEx Express uses a global air-and-ground network to speed delivery of time-sensitive shipments, usually in one to two business days – with the delivery time guaranteed.

Today FedEx has obtained a brand name of being the fastest delivery service with largest Air fleet through out the world. Here are some of the facts and figures about FedEx Express:

Headquarters	Worldwide: Memphis, TN. Asia Pacific: Hong Kong, China Canada: Toronto, Ontario Europe: Brussels, Belgium Latin America: Miami, FL
Principal Officer	David J. Bronczek, President and CEO
FY'03 Revenue	$ 16.5 billion
Average Daily Volume	More than 3.1 million shipments

Service Area	215 Countries
Workforce	Approximately 138,000 employees worldwide
Ground Fleet	Approximately 43,000 motorized vehicles
Air Fleet	638 aircrafts

Table 1-1 FedEx Express – Facts & Figures

1.2.2 FedEx Ground

Founded in 1985 as RPS; re-branded as FedEx Ground in 2000. FedEx Ground is North America's second-largest ground carrier for business-to-business, small-package delivery, with business-to-residential service available through FedEx Home Delivery

FedEx Ground specializes in cost-effective, small-package shipping, offering dependable business-to-business delivery or convenient residential service through FedEx Home Delivery. FedEx Ground guarantees delivery based on the shipping distance, usually in one to five business days in the United States and in three to seven days for select international locations. Some facts and figures about FedEx Ground:

Headquarters	Pittsburgh, PA
Principal Officer	Daniel J. Sullivan, President and CEO
FY'03 Revenue	$ 3.6 billion
Average Daily Volume	More than 2.1 million shipments
Service Area	Service throughout the United States, Canada and Puerto Rico

Workforce	Approximately 43,000 employees and independent contractors
Ground Fleet	More than 17,000 motorized vehicles

Table 1-2 FedEx Ground – Facts & Figures

1.2.3 FedEx Freight

FedEx Freight is the U.S. market leader in providing next-day and second-day regional, less-than-truckload (LTL) freight services. FedEx Freight is composed of two operating companies – FedEx Freight East and FedEx Freight West, offering premium regional and interregional service to virtually every U.S. ZIP code, as well as international. Both are known for exceptional service, reliability and on-time performance.

FedEx Freight East was founded in 1982 as Arkansas Freightways; later renamed as American Freightways in 1993. FedEx Freight West was founded in 1966 as Viking Freight. Some facts and figures about FedEx Freight:

Headquarters	Memphis, TN
Principal Officer	Douglas G. Duncan, President and CEO, FedEx Freight
FY'03 Revenue	$2.4 billion
Average Daily Volume	More than 56,000 shipments
Service Area	Service throughout the United States, including Alaska and Hawaii; international and offshore

	service to Canada, Mexico, Central and South America, the Caribbean and Europe, and from Asia
Workforce	Approximately 22,000 employees
Ground Fleet	More than 39,000 pieces of equipment, including about 8,600 tractors

Table 1-3 FedEx Freight – Facts & Figures

1.2.4 FedEx Custom Critical

Founded in 1947 as Roberts Cartage; became an expedited carrier, Roberts Express, in 1980; re-branded as FedEx Custom Critical in 2000.

FedEx Custom Critical is North America's largest time-specific, critical-shipment carrier, providing nonstop, door-to-door delivery in the continental United States, Canada and Europe. Its subsidiary, Passport Transport, also specializes in the transportation of valued cars. Some facts and figures about FedEx Custom Critical:

Headquarters	Green, Ohio
Principal Officer	John G. Pickard, President and CEO
Average Daily Volume	More than 700 shipments
Service Area	Service throughout the United States and Canada, and within Europe
Workforce	Approximately 2,800 employees and contractors

Ground Fleet	More than 1,200 vehicles, ranging from minivans to tractor-trailers
Air Fleet	Access to virtually every on-demand aircraft in North America

Table 1-4 FedEx Custom Critical – Facts & Figures

1.2.5 *FedEx Trade Networks*

As the largest-volume customs entry filer in the United States, FedEx Trade Networks also provides a wide range of international support services to help make global shipping easier for customers of all sizes.

FedEx Trade Networks oversees two subsidiaries

- FedEx Trade Networks Transport & Brokerage Inc.,
- FedEx Trade Networks Trade Services Inc.,

Some Facts about FedEx Trade Networks:

Headquarters	Memphis, TN
Principal Officer	G. Edmond Clark, President and CEO
Service Area	Worldwide, from more than 330 locations
Workforce	Approximately 3,400 employees

Table 1-5 FedEx Trade Networks – Facts & Figures

1.2.6 FedEx Services

FedEx Services coordinates sales, marketing and technology support for the global FedEx brand, providing customers with a convenient, single point of access to FedEx shipping, information and supply chain services. FedEx Services has approximately 9,000 employee and this division has been recognized as "100 best places to work in IT" by Computerworld

2 Problem Statement

The Executive Sales Management division of FedEx has initiated various projects to report the current and future state of the customer by using financial/statistical analysis models. With the wide spectrum of existing reports, the user has to go through numerous reports to gather the required information and has to compile manually to see the track record of his/her clients.

In view of the above, the Executive Sales Management would like to enhance sales effectiveness by having visibility into the growth and decline of accounts across the sales organizations. This would enable the team to quickly identify growing or declining accounts and accordingly take pro-active and remedial action. The team would also like to have visibility into the growth of the sales responsibility areas (division/group/organization/region/district and territory) associated with the respective sales levels from CEO to the Sales Professional.

Currently there is no intelligent process to report on the growing or declining accounts. The process being followed is manual, which is time consuming and does not lead itself to flexibility in reporting. The visibility into the growth (whether positive or negative) of a sales responsibility area is also not available all the way up to the CEO level.

3 Objectives

Looking at the problem faced by the sales representatives, a solution has been proposed by the Sales Executive Team to address this issue. An automated report shall be generated to the user over the company's intranet for all the employees right from the sales representative to the CEO level.

The proposed solution is to create a report, the Growth & Ranking Report, which shall:

- List a pre-selected number of top gaining or declining accounts by either the actual change or the percentage change in revenue or volume or yield over a selected period would be the Growth. This would comprise the ***Ranking*** part of the report and will be called ***Gainers and Decliners***.
- Provide a roll-up of revenue or volume or revenue/volume or revenue/customers for the sales responsibility areas that come under a particular sales management level. This would comprise the ***Growth*** part of the Report and will be called the ***Territory Report***.

Note: FedEx has proposed the name Gainers and Decliners. The term Gainers shall mean the top 10 ten customers who have generated highest revenue in the period selected. Similarly, Decliners are the bottom 10 customer who have generated lowest revenue in the period selected.

Target Audience:

The target audience for this report would be the entire chain in the sales hierarchy (Sales Professional to the CEO level) and their Support Staff (Sales Analysts and Coordinators) for the following sales organizations:

- Inside Sales
- Field Sales
- International/Freight
- Corporate and Worldwide Sales
- Major Accounts

4 Data Warehouse Design and Implementation

The database is designed in a phased approach and the objective of this project is to implement the first phase roll out of the database creation and generation for the reports required. Thought the project is to design and develop a database, I have used the project management approach at all stages to implement the system.

The whole project was divided into small manageable pieces in order to pay special attention to every part and integrate the complete work to build a data warehouse for the reporting needs.

The project was guided with various controlling tools like Gantt chart and change management process. In the process of developing a Data Model, business user has requested for couple of changes and the Data Model has been updated accordingly. As the development was in the initial stage, there was no much impact of these change requests on the project time line and schedule. I could incorporate all the requests in the end deliverable according to the stipulated time line given by the client. The table below displays the project schedule:

Description	From	To	Duration
1. Requirements	7/1/2003	8/29/2003	44 Days
1.1. Understand the Business Process	7/1/2003	7/10/2003	8 Days
1.2. Collect the user requirements	7/11/2003	7/28/2003	12 Days
1.3. Document all the requirements	7/29/2003	8/27/2003	22 Days
1.3.1. Report	7/29/2003	8/13/2003	12 Days
1.3.2. Functional	8/14/2003	8/27/2003	10 Days
1.4. Get the documentation approved	8/28/2003	8/29/2003	2 Days
2. Planning/Design	09/01/03	10/16/2003	34 Days
2.1 Analyze source systems	09/01/03	09/12/03	10 Days
2.2 Request the data from source systems	09/13/03	09/21/03	5 Days
2.3. Create Data Models	9/22/2003	10/16/2003	19 Days
2.3.1. Conceptual	9/22/2003	10/2/2003	9 Days
2.3.2. Logical	10/3/2003	10/12/2003	6 Days
2.3.3. Physical	10/13/2003	10/16/2003	4 Days
3. Development	10/17/2003	10/26/2003	6 Days
3.1. Transform the physical data model	10/17/2003	10/22/2003	4 Days
3.2. Test the database	10/23/2003	10/26/2003	2 Days
4. Perform & Obtain User Acceptance Test	10/27/2003	10/31/2003	5 Days

Table 4-1 Project time line and Schedule

The Growth & Ranking is developed from scratch and there is no one source of information available to generate the report. The data required for the report shall be stored in a data warehouse by collecting data from various valid sources. This section gives an overview of the technologies used in developing the framework for the report.

4.1 Data Warehousing

Most of the organizations are "Data rich, but information poor" this happens when the organization is not utilizing its valuable data repository. In this competitive world, studying the pattern of a customer and acting accordingly is a key strategy to meet the cutthroat competition. Most of the organizations implement a Data Warehouse as a solution to the competition by analyzing the data using various statistical, analytical tools.

A Data Warehouse is a central repository for the data collected by the computer systems throughout the enterprise. Typically, a data warehouse is hosted on an enterprise's mainframe server, but can reside with any storage service provider. The success of a data warehouse strongly depends on the amount of qualitative resources the enterprise is willing to put into it. In the case of FedEx, Teradata is used to store the data for the complete organization.

The source of data for a data warehouse is collected through various online transaction processing (OLTP) applications and other sources are selectively extracted into a data warehouse. Once the data is stored into a data warehouse, various tools such as, online analytical processing (OLAP) tools, data mining,

statistical modeling, geographical information systems (GIS), decision support systems (DSS) applied to the repository to draw some valid conclusions and take a strategic decision.

There are five major elements of data warehousing

a. Data Acquisition

   Data Acquisition is a process of identifying, capturing, and transforming data in operational systems so the data can be loaded into a data warehouse or a data mart. It is complex, time-consuming, and costly phase in building a data warehouse. If this phase is not given an enough attention, the organization may face a huge problem when an analyst takes a critical decision based on the invalid data and then putting the company in jeopardy

b. Data Modeling

   Data Model is a blue print of the database, which will be implemented in future. Data Modeling is the process of analyzing the current business needs and designing a data model fitting the requirements. A data model consists of Entities, Attributes and relationships between the entities.

c. Metadata

   Metadata is defined as a description of the data. Most commonly, it is called as "data about data". Metadata simplifies the process of managing all the process involved in creating and sustaining a data warehouse. Metadata speak about the data and help a DBA to understand the existing or any new databases.

d. Data Management

Data Management includes the access and storage mechanisms that support the data warehouse. A good data warehouse implementation is designed to support complex queries. It is usually a relational, multidimensional, or other specialized database. A relational database is a collection of rows and columns organized as a set of formally described tables from which the data is access in many different ways without having to reorganize the database tables.

e. Data Analysis

The ultimate advantage of implementing a data warehouse is to support the decision support systems by performing data analysis. Data analysis applications enable end users to access and analyze data stored in the data warehouse or data marts. Data mining is the analysis of data for relationships that have not previously been discovered. It enables users to identify patterns and relationships within a set of data and create models to anticipate behavior or events based on trends in the data.

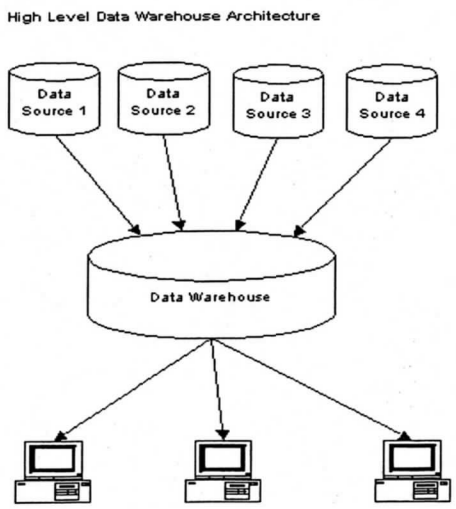


Figure 4-1 High Level Data Warehouse Architecture

4.2 Teradata

In order to support the data integration and storage of billions of records, FedEx has opted with its existing data warehouse, Teradata. Teradata offers bundle of benefits on implementation and is integral with other source systems and platforms. Teradata offers a full suite of data access and management tools, plus world-class services. The result: "a unique combination of products and expertise that provides a proven data warehousing solution"

Designed for decision support and parallel implementation, the Teradata Database is not constrained by the limitations that traditionally plague transaction-processing databases when they are used for data warehousing. Instead, the Teradata Database easily and efficiently handles complex data requirements and simplifies management of the data warehouse environment.

This is done by an automatic distributing data and balancing workloads. The Teradata Database's ease of use, combined with true query parallelism, allows it to achieve query performance and throughput levels unmatched by any other relational database management system (RDBMS).

Top 10 advantages of using Teradata to store the enterprise wide data:

a. Product Maturity

   This product is well developed by the engineer and has tackled all the technological glitches and their customer need not worry about technological issues and can concentrate on the strategic business value from the investment.

b. Reference Account

Teradata has a track record of 20 years in successful databases and is supported by impressive list of customers-companies who are leaders in their industries and in the use of data warehouse technology.

c. Quickest time to solution

With the features available in Teradata, the initial implementation and extensibility of the existing will be easy when compared to other products. The architecture of the Teradata has been inherited from parallel architecture and the self-managed feature gives the user flexibility in his/her implementation.

d. Lowest total cost of ownership

The largest Teradata site can be easily manageable with as few as two DBAs and these DBAs can extend their help to the business users to gather required information from the data warehouse rather spending all their time in restructuring data or tuning SQL queries to optimize the performance.

e. Complete Support Infrastructure

Teradata's support specialists are dedicated exclusively for data warehousing and have a very good expertise in the subject. These specialists offer their services round the clock and seven day a week.

f. Effortless Scalability

The real secrets behind the Teradata's scalability are:

- Unconditional parallelism
- Automatic hashing for even data distribution.

Teradata's virtual, shared-nothing database architecture enables automatic scalability that spans from single CPU servers through many-node massively parallel systems with no program changes or data restructuring. With Teradata, there are no sequential processing bottlenecks.

g. High User Concurrency

With the Teradata data warehouse, a user need not chose between the performance and concurrency as Teradata's scalability lets you maintain industry-leading performance no matter how huge the load is applied on the data warehouse.

h. Complex and Ad Hoc Query Performance

With the parallel processing, self-managing hashed data distribution, it provides the most advanced ad hoc and complex query environment on the market.

i. Fast, Fail-Safe data Utilities

Teradata ensure the mission critical availability of the business information with out interrupting the data access by other users. It has a full checkpoint restart capability and offers fully automatic parallelism of excellent performance.

j. Seamless Mainframe Integration

Teradata offers bi-directional high-speed channel connectivity to leading mainframe environments. It even offers seamless interoperability with the client platform environments.

4.3 Methodology

The database design process has undergone the process of Project Management and has milestones to cross at every stage of implementation. The first step in the

process is to gather requirements. During this step, I have interviewed prospective database users to understand and document their data requirements. The result of this step is a set of users' requirements. In addition to the data requirements, I have gathered functional and report requirements to better understand the usability of the gathered requirements.

After all the requirements have been collected and analyzed, I have created a conceptual schema for the database, using a high-level conceptual data model. This is otherwise called as a Conceptual design. The Conceptual design is a concise description of the data and user requirements. As this model does not include implementation details, they are easier to understand and are used to communicate with non-technical users.

Once the conceptual design is approved by the business user, it will be mapped onto the logical database design where the ER diagram will be transformed to the tables and columns by following the normalization process. Later this logical data model is converted to physical model by giving the actual table names and columns. At this stage, the table names are assigned according to FedEx standards and specifications. The following sections will discuss in detail about the design and implementation.

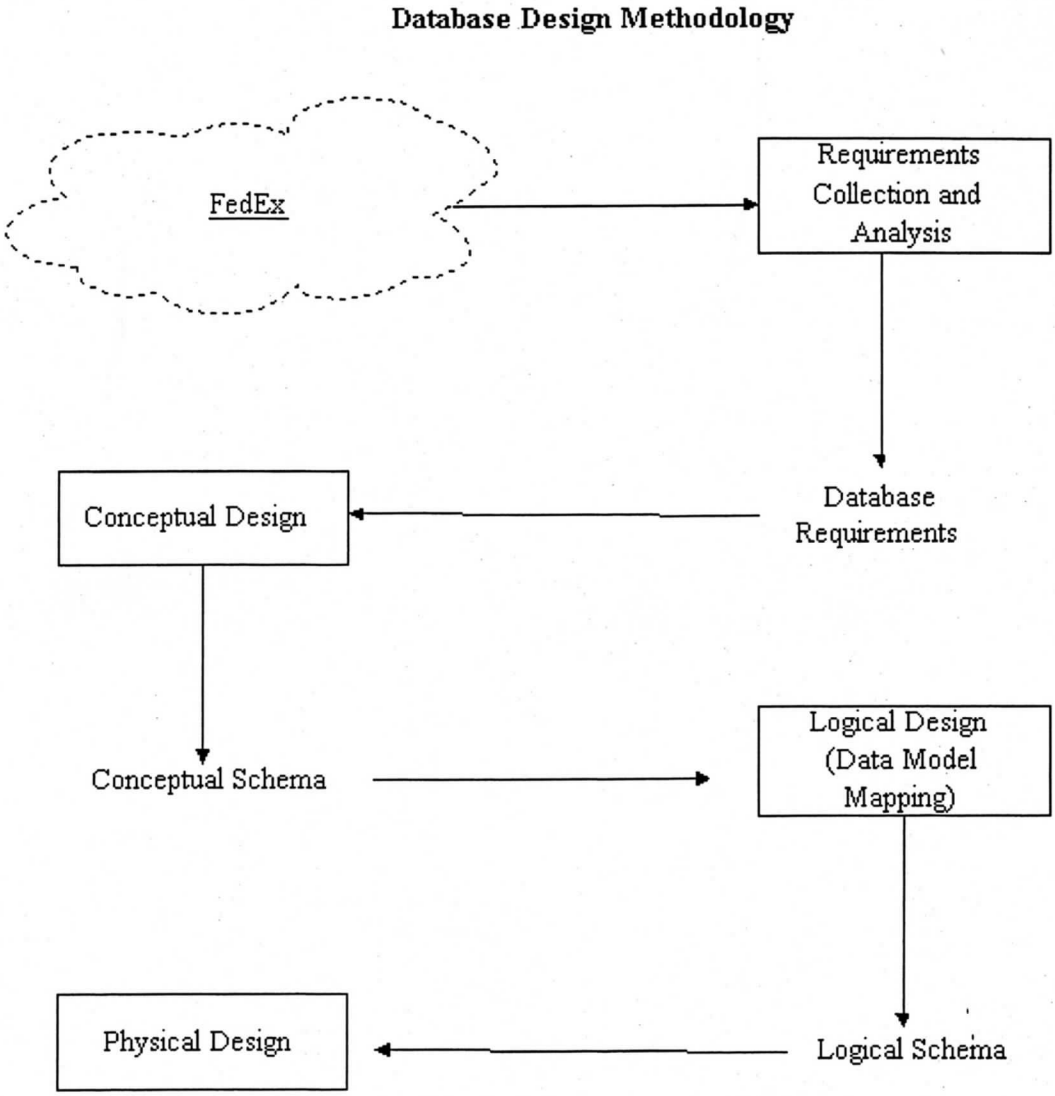


Figure 4-2 Database Design Methodology

4.4 Requirements collection and analysis

The requirements for the report have been gathered and are analyzed to extract the business rules for developing a Data Model. Here is the detailed list of business requirements

At a high level, the Growth & Ranking Report is to achieve two purposes:

1. Rank top 50 increasing or decreasing accounts by either the actual change or the percentage change in revenue or volume or yield. This would comprise the Ranking part of the report and will be called Gainers and Decliners.
2. Display the growth of certain sales indicators (ex. revenue or volume or revenue/volume) at a particular sales responsibility area compared over certain period of time intervals. This would comprise the Growth part of the Report and will be called the Territory Report

4.4.1 Gainers/Decliners (Ranking Report)

4.4.1.1 Business Requirement (Overall Description)

- The report shall have 2 views:
  - Summary view
  - Detail view
- The report shall be available to the following sales organizations:
  - Inside Sales
  - Field Sales
  - International/Freight
  - Corporate Accounts

- Major Accounts
- World Wide Services

- The report shall be available to all levels in a particular sales hierarchy. The roll up has to extend all the way up to the CEO.
  - Corporate Accounts:

    CAE -> CAM -> CAD -> VP -> SVP -> CEO
  - Major Accounts

    MAE -> MAM -> MAD -> VP -> SVP -> CEO
  - Inside, Field Sales, WWS, International/Freight

    AE -> DSM -> RSD -> VP -> SVP -> CEO
- The roll-up of the report to any level above the sales professional (AE/MAE/CAE) up to the CEO level shall map to the real-life sales org setup. The roll-up logic should not be hard coded.
- The Sales Support Staff should be provided the same view as the managers to whom they directly report.
- SP&A staff shall be provided with full access for all levels (SP to CEO).
- Access to the reports shall be role and organization based. Depending on the Sales organization to which the user belongs, the user login shall determine the assignment

4.4.1.2 Business Requirement (User Interface)

- The report layout shall have tab buttons to allow a user to toggle between Gainers, Decliners and Graphs reports as desired by the user.

  See Appendix A.1

- The Gainers and Decliners global navigation elements/buttons shall be consistent across all levels of the sales hierarchy. These elements shall be:

  See Appendix A.1

Field Name	Description	Method	Default Field Display
Account Type	Ability to rank 9-Digit, 5-Digit	Drop Box	Display the specific account type applicable to the user. If a user has more than one account type responsibility, such rankin should default to display *9 Digit* in the field
Service Category	Ability to rank the report by either Express, Ground, International or Overall (sum of all the three)	Drop Box	Display *Express* in the fiel
Measure Type	Display report by either Revenue or Volume or Yield	Drop Box	Display *Revenue* in the field
Period	Ability to choose either: a) WOW b) MOM c) QOQ	Drop Box	Display *WOW* in the display field

Field Name	Description	Method	Default Field Display
	d) YOY e) Last Three Months Avg. vs. Current Month		

Table 4-2 Global Navigation

- The report shall provide the user with the following single button click:
  - Download
  - Excel Summary View (Radio Button)
  - Excel Detailed View (Radio Button)
  - PDF Summary View (Radio Button)
- The user can make his selection by clicking the appropriate button. This automatically shall allow the user to view the report as per the filter selections made by him/her (the user would need to click on the refresh button/tab to get the desired view).
- The screen colors, navigation and font family shall be defined by the sales.fedex.com Style Guide guidelines

4.4.1.3 Business Requirement (Common Functionality)

The Gainers and Decliners Report shall rank the gaining or declining accounts. The ranking shall be based on either the actual change or the percentage change in the measure types over the period selected by the user.

- The "measure type" field shall include revenue, volume or yield based on the filter selection made by the user.

- For calculation purposes,
  - The measure for revenue shall be Adjusted Net Revenue.
  - The measure for volume shall be total packages shipped.
  - The measure for yield shall be the yield per package.
- The Period of comparison available to the user shall be:
  - WOW
  - MOM
  - QOQ
  - YOY
  - Last Three Months Avg. vs. This Month
- The computation for the Last Three Months Avg. shall be based on:

  (Summation of the Measure Type values for the 3 months in consideration)/ 3
- At least 24 months of data shall be available to enable comparison between different time dimensions, especially for YOY comparison.
- The gaining and declining accounts/ organization shall be displayed in the descending order (i.e. the highest gainer or decliner shall be ranked first) by default. However, sorting functionality shall be provided to sort by ascending or descending order as desired by the user.

4.4.1.4 Business Requirements (Summary Report Functionality)

- The columns in the tabular body of the report shall have sorting functionality. The available sorting type is listed in the table below:

Sl. #	Column(s) Heading	Sort By
1.	Account Name	Alphabetical Order
2.	Organization Name (could be Sales Org or Region or District or Territory based on the user viewing the report)	Alphabetical Order
3.	Sales Professional	Alphabetical Order
4.	Date Last Called	Descending Dates (nearest date to farthest date) or Ascending Dates (farthest date to nearest date)
5.	Period <base> <comparison>	Ascending or Descending Order of values
6.	Actual Change	Ascending or Descending Order of values
7.	% Change	Ascending or Descending Order of values

Table 4-3 Sort Functionality

- The user shall have the capability to drill down from the 5-Digit to the corresponding 9-Digit account ranking. This drill down capability should be provided by some visual representation so that a user can know that a particular 5-Digit has 9-Digit accounts associated with it.
- The viewed reports shall also be available for download. The following shall be provided:

- Excel Download Summary View-This shall download the current report exactly as it is available online
- Excel Download Detail View-This shall download the detail view of the current report being viewed. All the filter criteria selected for the summary view shall be retained.
- PDF Summary View- This shall download the report as it appears on the screen.

  Note: Irrespective of the download format selected, the column headings in any of the downloaded formats shall exactly match the column headings in the report being accessed online. The Excel version of the report shall download with all standard MS Excel functionality to aid a user in his/her analysis of the data in the report. User shall be able to sort and filter on the data downloaded.
- The Summary view of the report shall provide the following information:

Main Column Heading	Sub Column Heading	Description	Special Notes
Account #		Could be 9 Digit or 5 Digit depending on filter chosen	Availability of 9-Digit, 5-Digit optio shall depend on the role and access rights defined at login
Account Name		Name of the account being selected	
Organizatio		For CEO Level:	The Organization Name column will

Main Column Heading	Sub Column Heading	Description	Special Notes
n Name		Shows all the Sales Orgs to which the ranked accounts belong. For SVP Level: Shows all the Sales Orgs to which the ranked accounts belong For VP Level: Shows all the Regions to which the ranked accounts belong For RSD Level: Shows all the Districts to which the ranked accounts belong For DSM Level: Shows all the Territory #s to which the ranked accounts belong	not be available for the SP level

Main Column Heading	Sub Column Heading	Description	Special Notes
Sales Professional		Name of the Sales Professional who is assigned to the account	Representation of data: First Name Last Name
Date Last Called		Date on which the account was last called	Representation of data: MM/DD/YY
<Service Category>		Service Category depending on filter chosen	Express or Ground or International o Overall
	<TD1> <base period>	Time Dimension Measure depending on option chosen for Period field (could be WOW or MOM or QOQ or YOY or Last Three Months Avg. vs. Current Month) Measure Type: Revenue or Volume or Yield	Representation of column heading: For WOW: <Weekend date> <Measure Type> Ex: 10/20/2002 Revenue For MOM: <Month Name> <YY> <Measure Type> Ex: Oct 02 Revenue For QOQ: <Quarter Name> <YY>

Main Column Heading	Sub Column Heading	Description	Special Notes
			<Measure Type> Ex: Q2 02 Volume For YOY: <Month> <YY> <Measu Type> Ex: Oct 02 Volume For Last Three Months Avg: Avg.<M1><M2><M3><YY> <Measure Type> Ex: Avg. Mar/Apr/May 02 Revenue
	<TD2> <Measure period>	Same as the column above	Representation of column heading: Same as shown for the sub-column <TD1>
Actual Change		Relative Change in Measure Type depending on filter	

Main Column Heading	Sub Column Heading	Description	Special Notes
		chosen	
% Change		% Change in Measure Type depending on filter chosen	Rule: The formula for calculating percentage change shall be the conventional formula: [(Change in measure being calculated)/original measure] X 100 For cases where the % change has to be calculated between two time-bound instances of the measure of which the original measure is zero or tending towards zero (example: a new account being acquired or a dormant account giving business), such a calculation would be left blank and the rule will be to represent that % change by a hyphen (-).

Table 4-4 Summary View Columns

Note: The representation of any data in the columns shall follow the rules given below:

For Revenue: Prefix amount by $

For Yield: Prefix amount by $

Numerical values and percentage values shall be rounded off to two decimal places

- The default Summary Report view filters shall comprise the following:

Field Name	Default View
# Of Accounts	Show either the top 50 gaining or bottom 50 Declining accounts as per the toggle tab chosen
Account Type	Rank by the specific account type applicable to the user. If a user has more than one account type responsibility, such ranking should default to the 9-Digit accounts
Service Category	Rank by Express
Measure Type	Display by revenue
Period	Display by WOW

Table 4-5 Summary Report Default Values

- Drill down functionality shall be provided on the Summary View depending on the User Class accessing the report:

Column Heading	User Class	Action on Clicking the Link	Special Notes
Account #	ALL	Display the Default Trend Report for that Account # from IR (Customer Report).	Account # in each row in the report should be hyperlinked
Date Last Called	ALL	Display the default Call History for that account (to be pulled from SAM: Accounts → Appointments)	Date Last Called in each row in the report should be hyperlinked
Organizatio	CEO	Displays the ranking for that Sales Org on basis of the same filter criteria as was chosen to initially rank the report. Further click-through leads to next levels: Region → District → Territory #	The Sales Org name in each row in the report should be hyperlinked The new display has to be in the same window.

Column Heading	User Class	Action on Clicking the Link	Special Notes
n Name	SVP	Displays the ranking for that Sales Org on basis of the same filter criteria as was chosen to initially rank the report Further click-through leads to next levels: Region → District → Territory #	The Sales Org name in each row in the report should be hyperlinked The new display has to be in the same window.
	VP	Displays the ranking for that Region on basis of the same filter criteria as was chosen to initially rank the report Further click-through should lead to next levels: District → Territory #	The Region name in each row in the report should be hyperlinked. The new display has to be in the same window.

Column Heading	User Class	Action on Clicking the Link	Special Notes
	RSD	Displays the ranking for that District on basis of the same filter criteria as was chosen to initially rank the report Further click-through should lead to next levels: Territory #	The District name in each row in the report should be hyperlinked The new display has to be in the same window.
	DSM	Displays the Gainers and Decliners ranking for that Territory # on basis of the same filter criteria as was chosen to initially rank the report	The Territory # name in each row in the report should be hyperlinked

Table 4-6 User Class Access

- The Sales Professional's name in the summary view of the report shall be hyperlinked. A click on the hyperlink should result in either a default mail

client window or a message box (with the sales professional's email id) being opened so that the user can send an email to the sales professional.

- The detail view of the report available on download shall comprise all the columns that are available for the summary view. The detail view should download in four separate sheets in the same Excel view:
  - Sheet 1: Rank accounts by Service Category chosen for the Summary view. (i.e. If Express is chosen as the Service Category in the Summary View, the Detail View should also rank by Express in Sheet 1).
  - Sheet 2: Rank the same # of accounts by the second Service Category (say Ground). The other parameters remain the same as in the Summary View.

    Note: The ranking could display account names different from those in Sheet 1.
  - Sheet 3: Rank the same # of accounts by the third Service Category (say International). The other parameters remain the same as in the Summary View.

    Note: The ranking could display account names different from those in Sheet 1.
  - Sheet 4: Rank the same # of accounts by Service Category chosen in the Summary View. However the report to also include the column headings and related data for the other Service category.
- The report shall have the capability to provide bar graphs based on the user selection criteria.

- .The Name of the Graph shall appear at the center of the graph report layout. The naming nomenclature shall be:

  Top <#of Accounts> Gaining or Losing Accounts by <Measure Type> in <TD1>: Actual Change <TD2> vs. <TD1> (depending on the filter options chosen by the user).

  Example:

  Top 50 Gaining Accounts by Revenue in Dec 02: By Actual Change Nov 02 vs. Dec 02

  The Bar Graph shall display the comparison between two time dimensions chosen based on service category and measure type. The parameters for the bottom section shall be:

  X-Axis: Account Name

  Y-Axis: <Service Category><Measure Type>

  Bar 1: <Time Dimension 1> Color Code: Blue

  Bar 2: <Time Dimension 2> Color Code: Purple

4.4.2 Territory Report (Growth Report)

4.4.2.1 Business Requirement (Overall Description)

- The territory report shall be available to the following levels of Sales Management: CEO SVP, VP, RSD, and DSM. The report shall be role based.
- The report shall have roll-up capabilities. The roll-up logic shall not be hard coded.
- The sales support staff shall be provided the same view as the managers to whom they report. SP&A staff shall be provided with full access for all levels.

4.4.2.2 Business Requirement (User Interface)

Global navigation elements shall be available for all levels of sales management. The global navigation elements shall provide drop down capabilities. The options available in the drop down shall be role based. See Appendix A.2

- The elements shall be:
    - Organization
    - Region
    - District
    - Territory
    - Service Category
    - Measure Type

- Period

- The columns in the tabular body of the territory view shall have sorting functions. The available sorting type is listed in the table below:

SI #	Column(s) Heading	Column(s) Sub-heading	Sort By
1	Organization		Numeral Ascending or Descending order
2	Organization Name		Alphabetical Order
3	< Service Category>	<Time Dimension 1> <Measure Type>	Ascending or Descending order of values
		<Time Dimension 2> <Measure Type>	Ascending or Descending order of values
		Actual Change or % Change	Ascending or Descending order of values
4	Overall	<Time Dimension 1> <Measure Type>	Ascending or Descending order of values
		<Time Dimension 2> <Measure Type>	Ascending or Descending order of values
		Actual Change or % Change	Ascending or Descending order of values

Table 4-7 Territory View Sorting

- The viewed report shall also be available for download. The "Download As" option shall provide the user with PDF and Excel download options.

  Note: Irrespective of the download format selected, the column headings in any of the downloaded formats shall exactly match the column headings within the report being accessed online.
- The Excel version of the report shall download with all standard MS Excel functionality to aid a user in his/her analysis of the data in the report.
- Drill down functionality shall be provided on the Territory Report depending on the User Class accessing the report. The drill down capability shall be available on the Organization Name columns. The drill down would be a function of the role and entitlements of the User Class. See Figure 1-2
- The report shall have the capability to provide bar graphs based on the user selection criteria. The graphs would be role based.
  - The Name of the Graph shall appear at the center of the graph report layout. The naming nomenclature shall be:

    Territory Report for <Organization Name> by <Measure Type> in <TD1>: Actual Change or % Change <TD2> vs. <TD1> (depending on the filter options chosen by the user).

    Example:

    Territory Report for US East by Revenue in Dec 02: Actual Change Nov 02 vs. Dec 02

Territory Report for Mid-Atlantic Region by Customers in Q303: % Change Q203 vs. Q303

The Bar Graph shall display the comparison between two time dimensions chosen based on service category and measure type. The parameters for the bottom section shall be:

X-Axis: Organization Name

Y-Axis: <Service Category><Measure Type>

Bar 1: <Time Dimension 1> Color Code: Blue

Bar 2: <Time Dimension 2> Color Code: Purple

4.5 Conceptual Design

As the source of data for generating this report is from various origins, I have adopted the tiered approach in extracting the required data and as the tiers increase, the data is summarized and stored. Business rules are applied to each tier and are stored in the next tier. In the process of extraction, I have used three tiers to extract the summarized information. The tables used in their tiers are de-normalized to certain extent for the sake of performance.

4.5.1 Tier 1

There are five service categories available in the database and these have to be integrated into three service categories by joining International and International Next Flight, FedEx Ground and Sameday.

4.5.1.1 Domestic Express

The domestic express data is generated using two existing tables called dom_shipment and shp_x_spec_hand. These tables are populated into staging tables by applying some data specific business rules. At the end of this tier, a table called dom_exp_cust_rev_summary table is created in the development environment.

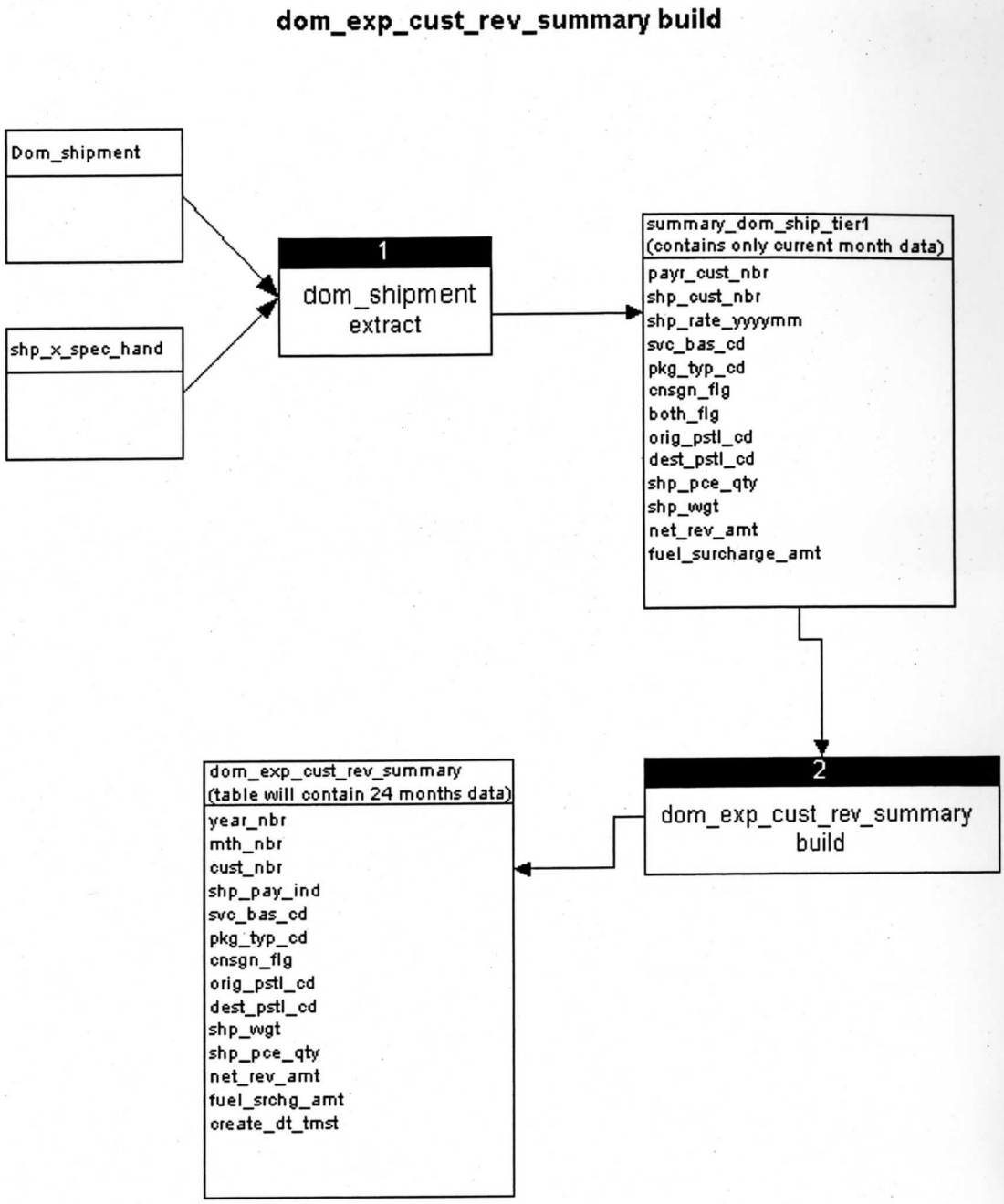


Figure 4-3 Domestic Express Customer Revenue Summary Build

4.5.1.2 FedEx Ground

The FedEx ground data is generated using two existing tables called rps_invoice and rps_inv_acs. These tables are populated into staging tables by applying some data specific business rules. At the end of this tier a table called fdx_gnd_cust_rev_summary table is created in the development environment

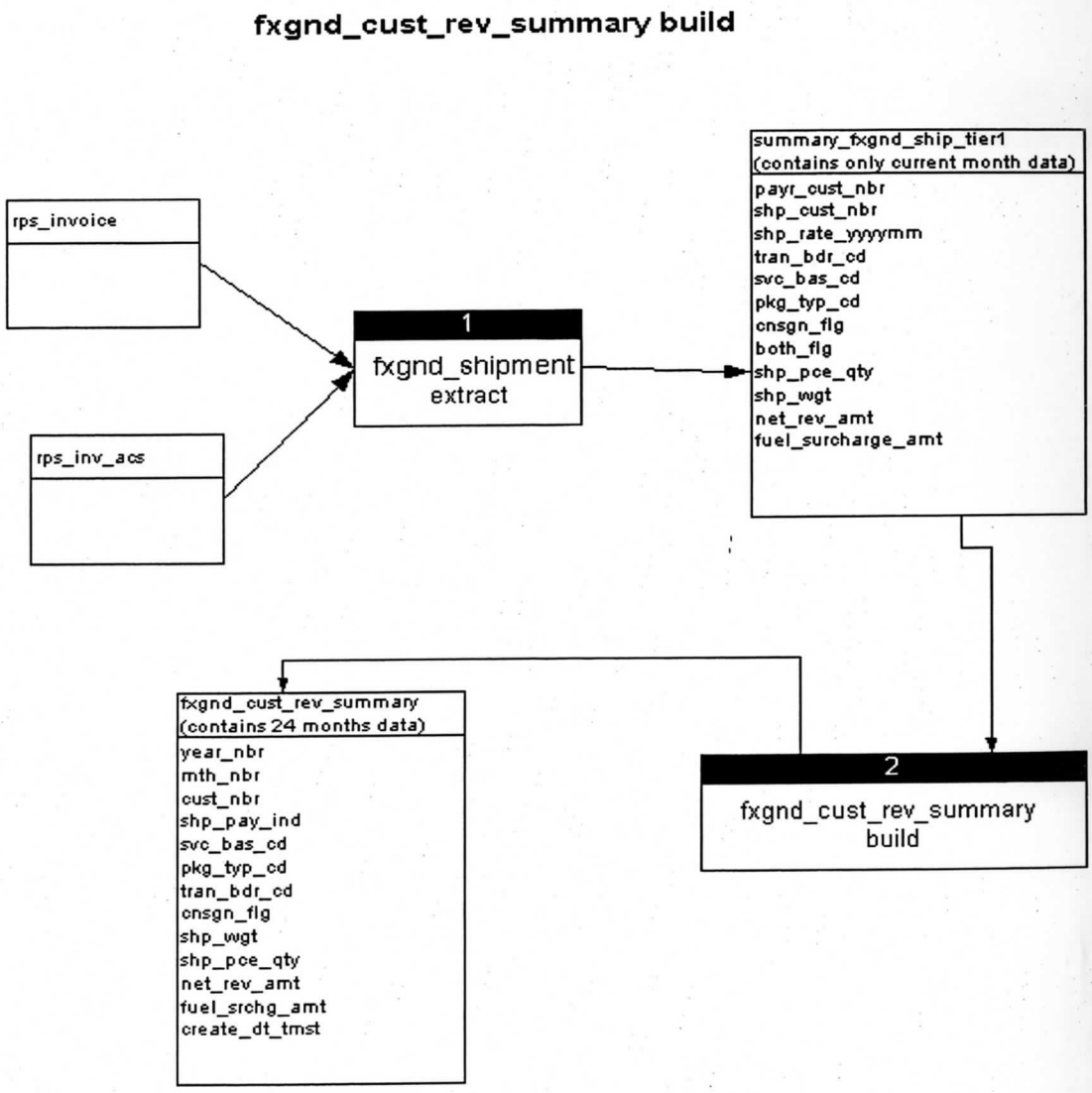


Figure 4-4 FedEx Ground Customer Revenue Summary Build

4.5.1.3 International

The International data is generated using two existing tables called intl_shipment and shp_x_spec_hand. These tables are populated into staging tables by applying some data specific business rules. At the end of this tier a table called dom_exp_cust_rev_summary table is created in the development environment

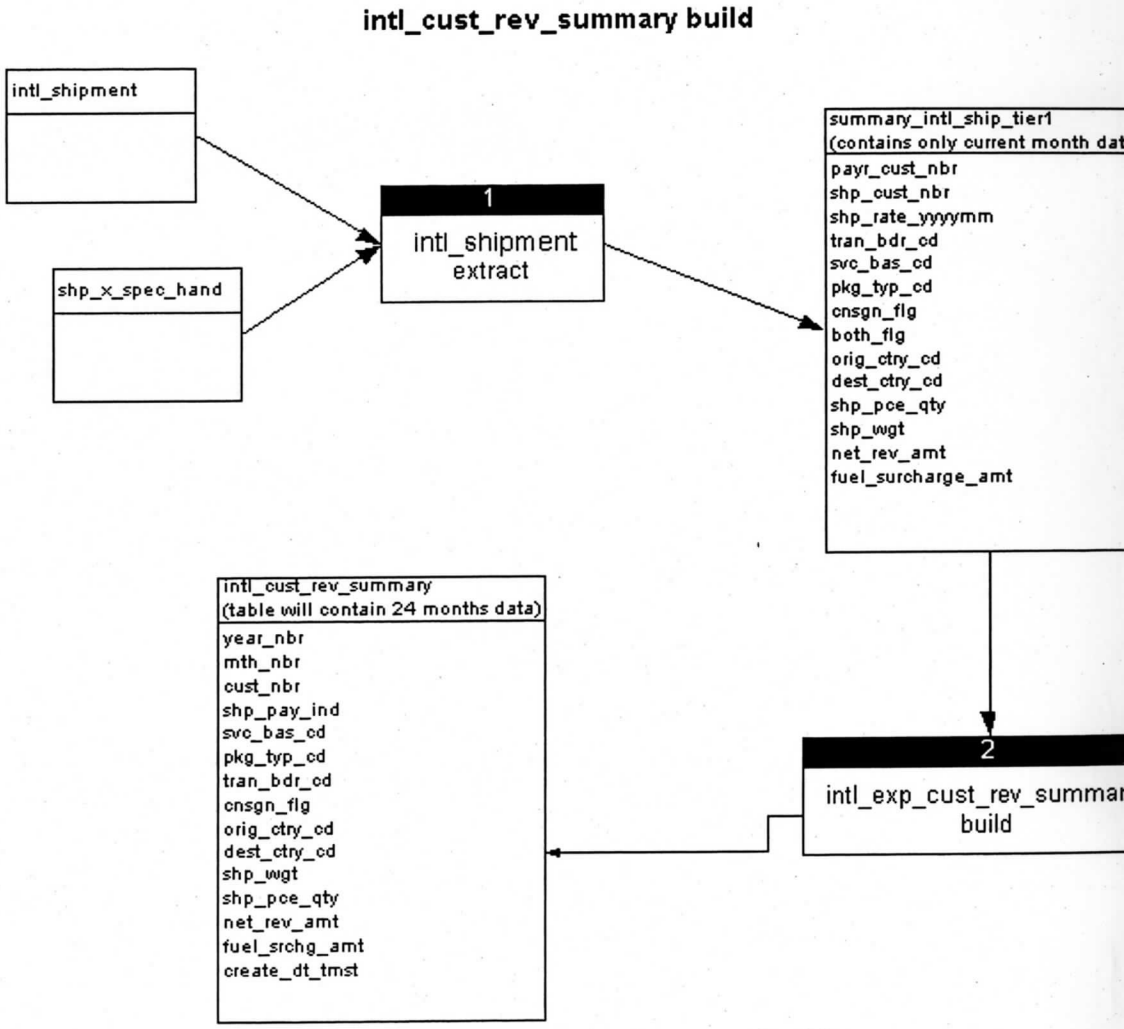


Figure 4-5 International Customer Revenue Summary Build

4.5.1.4 International Next Flight

The International Next Flight data is generated using two existing tables called intl_next_flg_shp and intl_next_flt_x_srchg tables. As the volume of data in these tables is considerably low, I have used only one staging table to export the table into a consistent data. At the end of this tier a table called intl_next_flt_cust_rev_summary table is created in the development environment

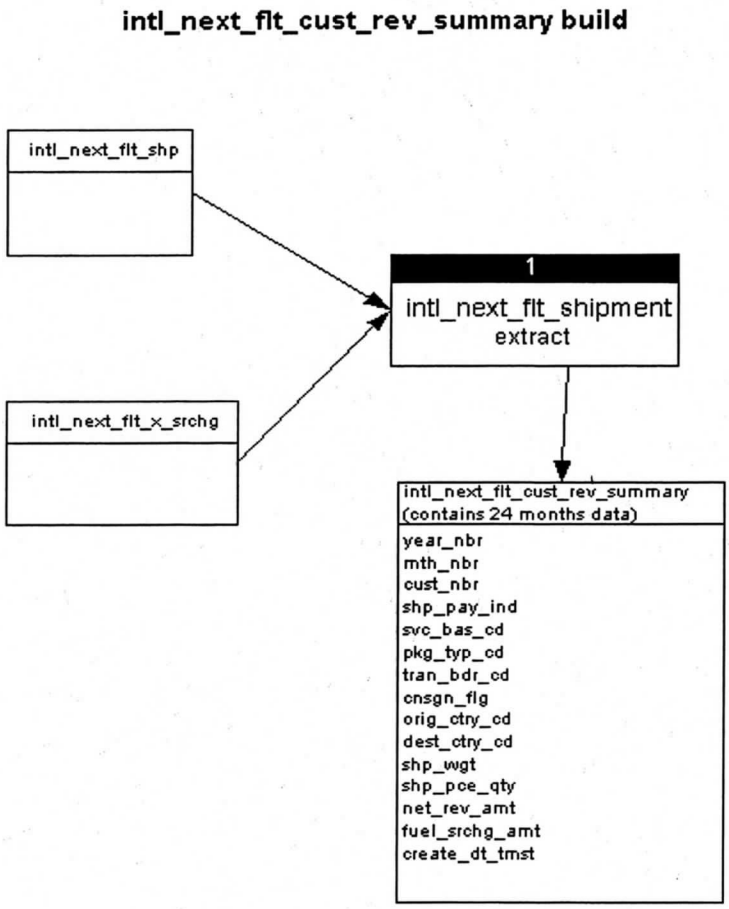


Figure 4-6 International Next Flight Customer Revenue Summary Build

4.5.1.5 Sameday

The Sameday data is generated using two existing tables called sameday_shp and sameday_shp_x_srchg tables. As the volume of data in these tables is considerably low, I have used only one staging table to export the table into a consistent data. At the end of this tier a table called sameday_cust_rev_summary table is created in the development environment

same_day_cust_rev_summary build

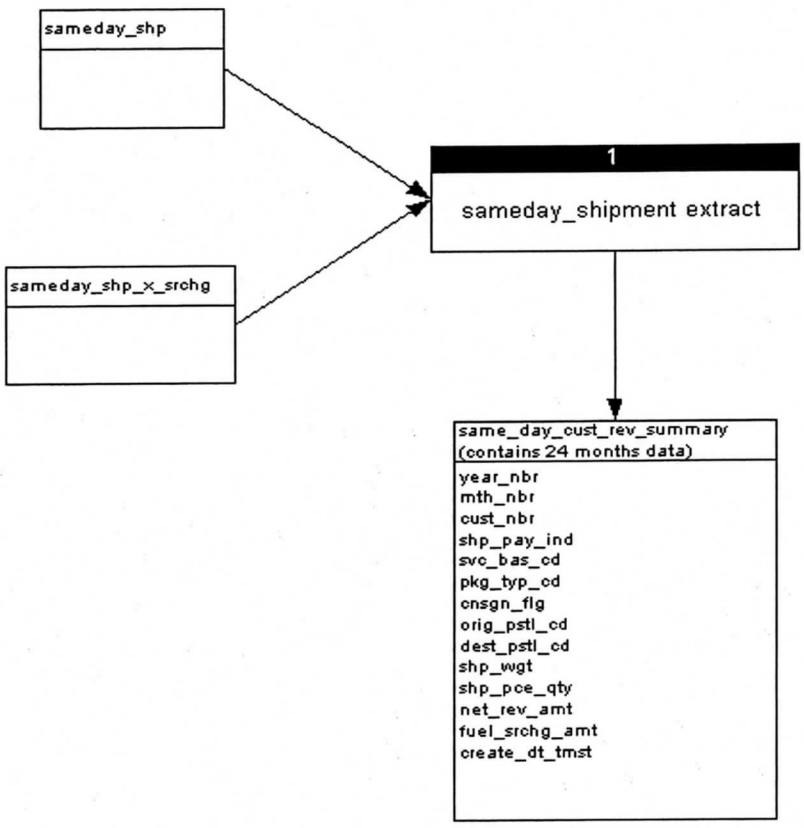


Figure 4-7 Sameday Customer Revenue Summary Build

4.5.2 *Tier 2*

In this tier, I have summarized all the five tables into one by adding a two new columns, svc_cat_cd and svc_bas_cd. These two columns combined together will determine the service category as either Domestic or International or Ground. Svc_category_xref acts as a domain table and has all the possible combinations of svc_cat_cd and svc_bas_cd. At this stage, the data in cust_rev_summary contains the revenue/volume generated by the customer irrespective of the employee who is assigned to that customer.

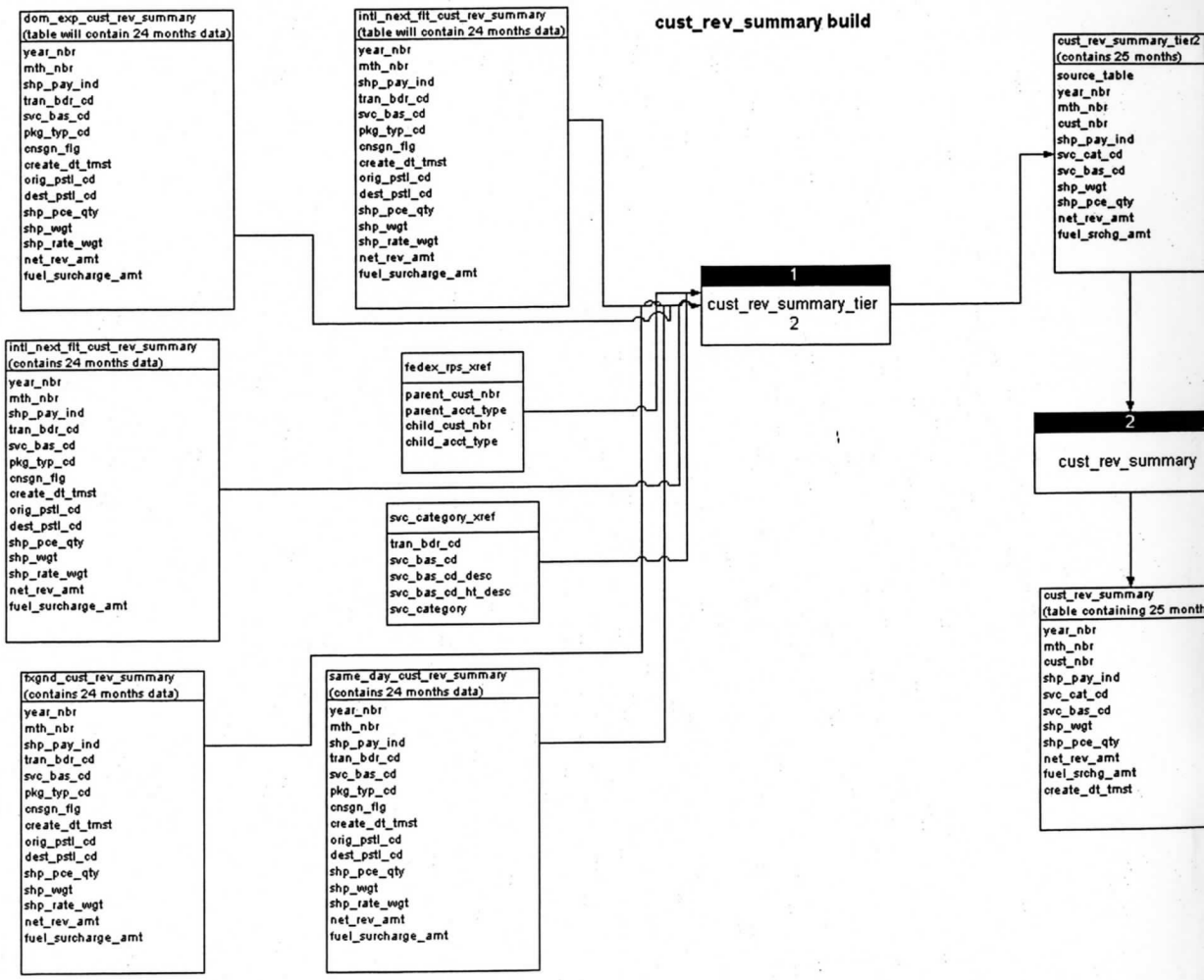


Figure 4-8 Tier 2

4.5.3 *Tier 3*

This is the final tier in the process of extracting the required data for the report. In this stage, I have created a common table that acts as a lookup table to the main revenue table. This lookup table is created with customer_alignment, employee_assignment, sales_call_interactios, Customer, national_acct. These tables give the information about the customer-aligned territory and employee's assigned territory. With the common element of territory number, a mapping has been established between the customer and an employee.

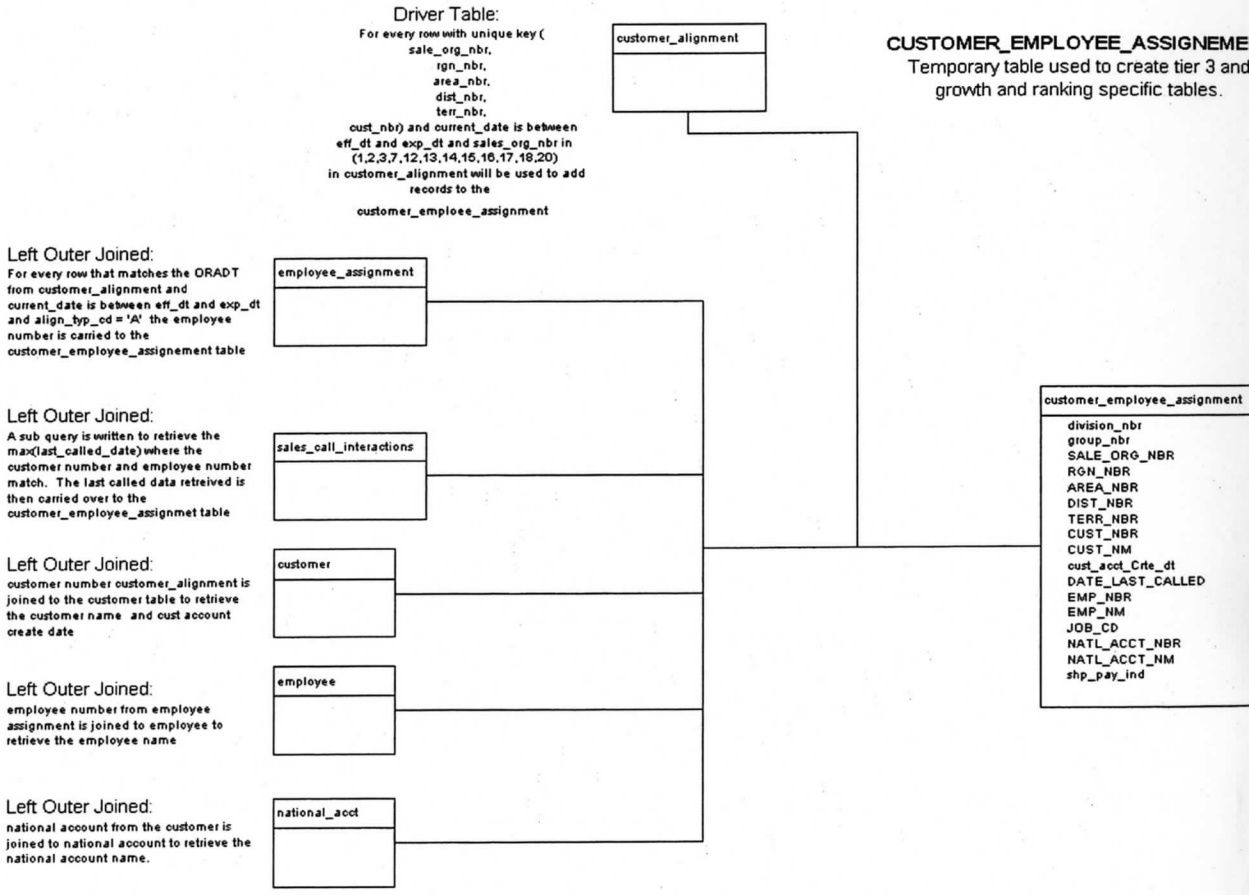


Figure 4-9 Tier3 – Supporting Table

The end table, customer_employee_assignment is used to create the final tier. This tier has summarized tables for both month and quarter. The data in the month's tables are summed together and has created the quarter table for the sake of performance and faster access of data.

The summarized tables shall store the 25 months of data to support the report requirements. The data in these tables follow the FIFO (First In First Out) policy by deleting the oldest data when a new month's data is ready to populate. The data is populated on the first Sunday of every month for the previous month. Source systems will upload their data into the base tables by the end of first Sunday of the current month. A batch process is run to populate the target tables.

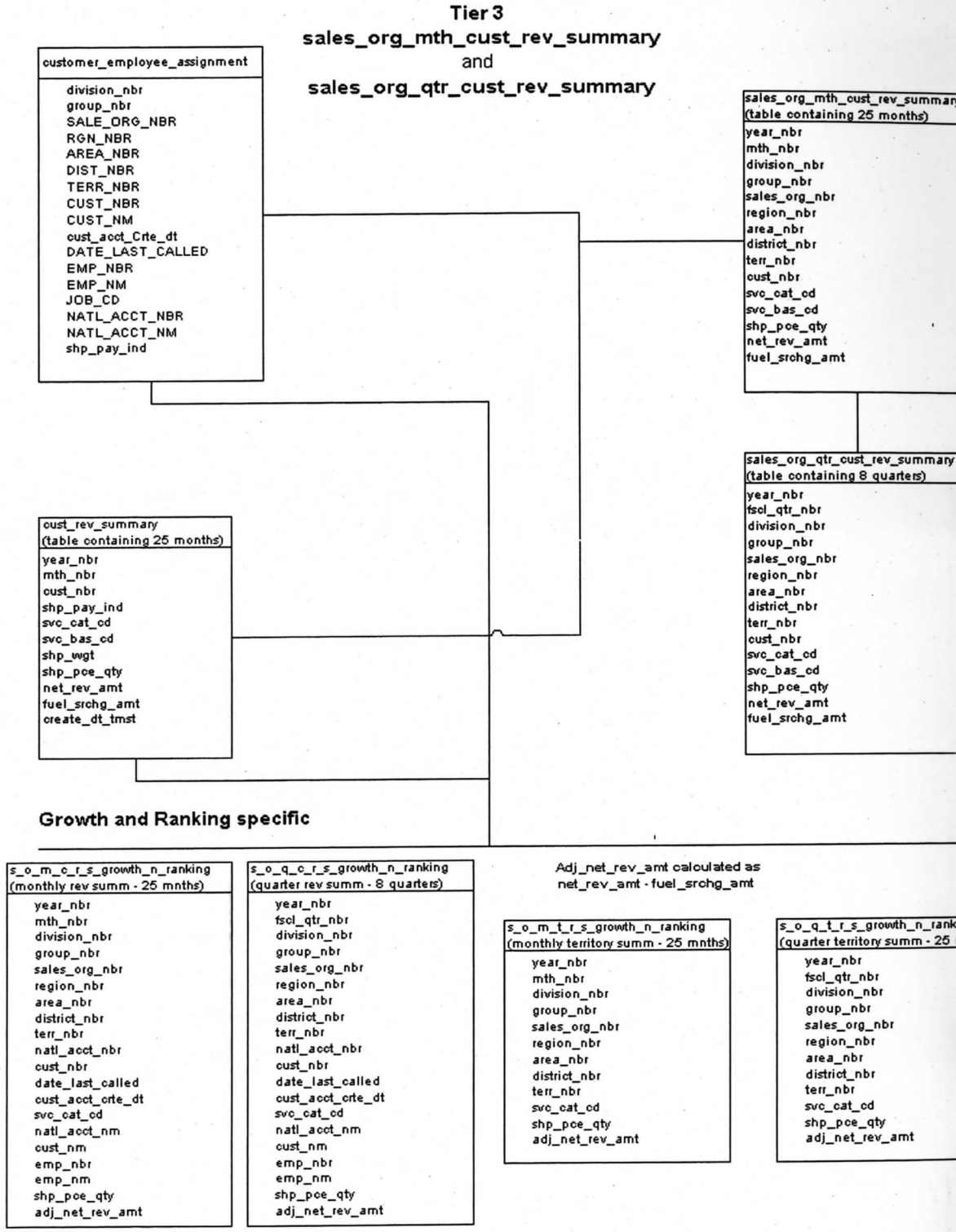


Figure 4-10 Tier3

4.6 Logical Design

In a normal procedure of creating a database design, this is the second phase after Conceptual Design. With the kind of data and the summarization of data into one single table, I have decided to create physical data model by eliminating this intermediate stage. Apart from the summarization, there was an extensive exercise during the conceptual design phase and the tables were designed in the same phase.

The utility of having a logical data model helps when the database has to be designed from scratch and the developer has liberty of selecting the entities and attributes based on the requirements. However, in this case of designing, I do not have an option of selecting the entities and attributes as the existing data drives the design and at the same time, I need to keep the business requirements in hand to validate the data

4.7 Physical Design

As discussed in the logical design section, I have created the physical data model right from the conceptual design. The Physical data model displays the actual table names and the column names, which will be implemented later in the production systems. This model displays the relationship between tables and these relationships are called as cardinality.

The Physical data model is a result of the physical design and is specific to the database on which it is implemented. With the huge amount of data (approximately two billion records) in these tables, I have opted for summarizing as much as possible. The criteria for summarization are Customer, Service category, period. This means that the revenue generated by a customer for a service category is summed up and stored in these tables with the weekend date, month number and year number. This summarization enables the faster access of the data and a low overhead on the database engine to compute the calculations. With the number of selections available for the user, it is not feasible to have a canned report and the other option is to generate the reports on the fly as per the user's selection. In order to support these computations, there will be a huge overhead on the database engine to compute for every combination of the user. To reduce this overhead, I have de-normalized the tables and summarized the data for easier and faster access of data.

A brief description on the tables in the Physical Data Model:

a. Customer_employee_assignment

   The objective of creating this table is to support the need to map a territory to a customer and at the same time to an employee. Territory number (12-digit combination of sales_div_nbr, sales_grp_nbr, sales_org_nbr, sales_region_nbr, sales_area_nbr, sales_dist_nbr, sales_terr_nbr) is the common element between an employee and a customer. Apart from mapping the relationship between the customer and an employee, this table stores information about the date on which a customer has been contacted by the employee and provides a relationship between the 5-digit account number and a 9-digit account number.

b. Employee

   This table maintains the list of all employees in the organization right from the sales professional to CEO. This table provides information about the Sales Planning and Analyst team who has an access to all the reports and the support staff of the senior management. The mgr_lvl_nbr gives the level to which a user is assigned.

c. Sales_territory

   This table provides the names of the division, group, organization, region, area and district for every combination of the territory number. This table acts a lookup table to populate the name of the division, group, organization, region and district. Sales_terr_nbr is the last two digits in the 12-digit territory number and is not assigned with a name.

d. Grow_rank_week_cust_rev_sum

This is a summarized end table for the gainer and decliners' part of the report. The summarization is for a customer who has generated revenue/volume in a period of a week and with the specific service category. This table shall store 25 months of data to compare the current year revenue with the previous year. In this case, a year shall be a rolling period. A rolling period shall start from the month before the current month to the year before. Example: If the current month is Oct 2003 the current year shall be Oct 2002 to Sept 2003. For this reason, the database has to maintain the 25 months of data.

e. Grow_rank_qtr_cust_rev_sum

This table is similar to the grow_rank_week_cust_rev_sum but the difference is that the revenue/volume is summed up for a quarter rather than a week to support the comparison of one quarter with other.

f. Grow_rank_week_terr_rev_sum

This table provides the required data for territory report. A territory report displays the revenue/volume generated in a territory rather than the revenue generated for a customer. So, this table is summarized based on the 12-digit territory number, period and service category. Revenue for the week is summed and stored along with the weekend date.

g. Grow_rank_qtr_terr_rev_sum

This table is similar to the Grow_rank_week_terr_rev_sum but the difference is that the revenue/volume is summed up for a quarter rather than a week to support the comparison of one quarter with other.

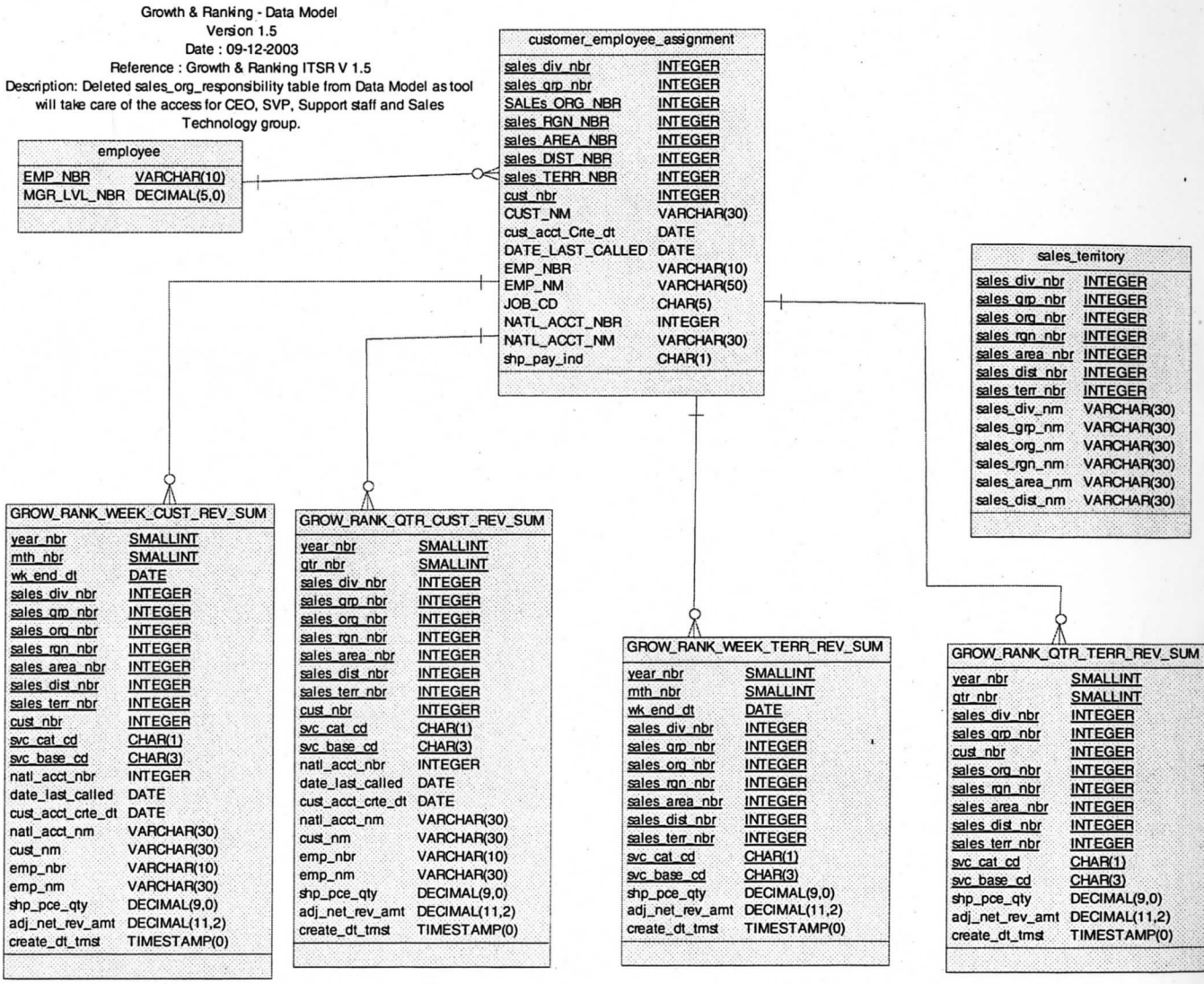


Figure 4-11 Physical Data Model

Note: Please refer to Appendix A.8 to A.14 for an enlarged tables

4.8 Implementation

Once the physical data model has been approved by the business user, next phase is to develop the required tables and load the required data into them. These tables are initially created on the development system to analyze the impacts and complexity on implementation. With the successful rollout on the development system, a batch process is executed on the production system to deliver it to the end user.

The implementation process is specific to Teradata Data Warehouse. The SQL generated is compliant with the Teradata semantics and cannot be executed on any other system. As discussed in the conceptual design phase, the implementation process shall undergo the tiered phases. In this process, data shall move from table to table by applying data rules at every stage.

See Appendix B.1 for the SQL code used to implement the tables on the Teradata data warehouse.

5 Future Enhancements

The designed database is based on the requirements for Growth & Ranking report and the data in these tables are highly specific to meet the needs. The ISH team has decided to enhance the design to accommodate the needs and requirements of the business user. As a next step to the design, the database will be enhanced and make the data in the data model to be flexible and support any new requirements.

Sales planning & analysis division has submitted a proposal to the ISH team for development for new reports. The current data model is designed to accommodate any future changes and is very adjustable.

6 Conclusion

The implementation of a data warehouse shall consider various aspects of the company and shall integrate with data from all the departments to provide sufficient data for analysis and data mining to foresee the problems and increase the customers' domain with better customer service.

A data warehouse shall provide intelligence to the marketing division with various statistical analysis and trends of the future business. A successful data warehouse implementation shall have a high Return On Investment, Competitive Advantage over the competitors, increased productivity of Corporate Decision Makers.

References

[1] Elmasri, R. and Navathe, S. B. (2001). Fundamentals of Database Systems. India: Pearson Education Asia

[2] Giovinazzo, W. A. (2000). Object-Oriented Data Warehouse Design. New Jersey: Prentice Hall PTR

[3] Jones, V. A. (2001). Data warehousing: A different type of central file. *Office Solutions.* Mt. Airy: Aug 2001. Vol. 18, Iss. 7; pg. 26, 5 pgs

[4] Garner, H. (2003). Data warehouses improve communications. *Community College Week*, Oct 13, 2003 v16 i5 pS5(2)

[5] Gardner, S. R. (1998). Building the Data Warehouse. *Communications of ACM.* September 1998/ Vol. 41, No. 9

[6] Website for reference: www.teradata.com

[7] Website for reference: www.sybase.com

[8] Website for reference: www.fedex.com

Glossary

#	Abbreviation/ Terms	Name
1.	MAE	Major Account Executive
2.	CAE	Corporate Account Executive
3.	MAD	Major Account Director
4.	CAD	Corporate Account Director
5.	MAM	Major Accounts Manager
6.	CAM	Corporate Accounts Manager
7.	SVP	Senior Vice President
8.	VP	Vice President
9.	RSD	Regional Sales Director
10.	DSM	District Sales Manager
11.	AE	Account Executive
12.	3-Digit Account	Grouping of 5-digit account numbers that are combined for the purposes of reporting and alignment to a territory; also called Super National Account
13.	5-Digit Account	Grouping of 9-digit accounts for the purpose of alignment, reporting, and compensation; earlier referred to as National or Global Account, now also called Corporate Account
14.	9-Digit Account	The account number that represents a single location or group shipping by FedEx
15.	Measure Type	The business measure that may be used to gauge sales effectiveness

#	Abbreviation/ Terms	Name
		(ex: revenue or volume or yield or revenue/customers or # of customers)
16.	Org	Sales Organization
17.	Service Category	The FedEx shipping service requested by the customer (could be either Express or Ground or International or a combination of any or all of the above mentioned service components)
18.	Period	The instance of time for which the report is being compared (ex. WOW, MOM, QOQ, YOY)
19.	Adjusted Net Revenue	Freight Charge + Special Handling Charge – Fuel Surcharge – Rebate – Discount
20	Volume	Sum total of all the packages shipped in a given period of time
21.	Yield	(Adjusted Net Revenue for a certain period of time) / (Volume shipped in that period of time)
22.	Avg.	Average
23.	YOY	Year on Year
24.	BR	Business Requirement
25.	User Class	The User Class is the generic term used to refer to the entire sales force (Sales professionals, Managers, Directors, VP and SVP)
26.	TD1	Time Dimension 1: The base time period (ex. if Period is MOM, TD1 could be Oct 02)
27.	TD2	Time Dimension 2:The time period that is to be compared to the base

#	Abbreviation/ Terms	Name
		time period (ex. if Period is MOM, TD2 could be Sep 02)
28.	MOM	Month on Month
29.	QOQ	Quarter on Quarter
30.	Period	The period the user wants to view. Month over Month, year over year, etc.
31	WOW	Week on Week

Table 7-1 Glossary

Appendix A.1 Summary View: CEO View

Summary View: CEO View

SVP Sales Org: All Organizations Period: MOM

VP Sales Org: All Organizations Measure Type: Revenue

Region: All Regions Account Type: 9-Digit

Districts: All Districts Service Category: Over All

Territory: All Territories

	Decliners	Graphs						
Acct #	Acct Name	Organization	Sales Professional	Date Last Called	Express Revenue ($)		Relative Change ($)	% Change
					Revenue Dec. 02	Rev. Nov. 02		
100078392	Sunoco Inc	US Sales	Michael Clarke	10/11/2002	$24,000	$35,000	$11,000	%50
108571969	Energy Argus	Corporate	Steve Smith	10/21/2002	$800	$9,800	$9,000	%40
133098429	Stanley Creations	US Sales	Pamela Lampkin	09/30/2002	$34,000	$42,500	$8,500	%10
258894766	Balchem Corp	US Sales	Carol Collins	11/17/2002	$21,000	$29,000	$8,000	%90
104229921	Balmar Inc	WW	Lori Till	11/02/2002	$5,600	$13,000	$7,400	%30
180639691	Square One Comm	US Sales	Richard Solmeo	10/08/2002	$11,000	$18,300	$7,300	%40
104297340	Babson College	WW	Peter Cercone	11/07/2002	$15,600	$22,000	$6,600	%10
114144258	Ameribag Inc	US Sales	Leonard Miret	10/25/2002	$16,500	$21,000	$4,500	%90
113412623	PayChex	Corporate	Brett Henry	10/10/2002	$40,000	$44,000	$4,000	%30
214596989	AMCO	WW	Samuel Stinson	09/22/2002	$14,000	$15,000	41,000	%30

Download: Excel Summary View | Excel Detail View | PDF Summary View

Appendix A.2 Summary View : Sr. VP – US Sales

Summary View: SR. VP US Sales

VP Sales Org: All Organizations Period: MOM

Region: All Regions Measure Type: Revenue

Districts: All Districts Account Type: 9-Digit

Territory: All Territories Service Category: Over All

Decliners | Graphs

Acct #	Acct Name	Organization	Sales Professional	Date Last Called	Express Revenue ($)		Relative Change ($)	% Change
					Revenue Dec. 02	Rev. Nov. 02		
100078392	Sunoco Inc	US Central	Michael Clarke	10/11/2002	$24,000	$35,000	$11,000	%20
108571969	Energy Argus	US East	Steve Smith	10/21/2002	$800	$9,800	$9,000	%30
133098429	Stanley Creations	US West	Pamela Lampkin	09/30/2002	$34,000	$42,500	$8,500	%40
258394766	Balchem Corp	Intl/Frt	Carol Collins	11/17/2002	$21,000	$29,000	$8,000	%80
104229921	Balmar Inc	Retail	Lori Till	11/02/2002	$5,600	$13,000	$7,400	%20
180639691	Square One Comm	US Sales	Richard Solimeo	10/08/2002	$11,000	$18,300	$7,300	%30
104297340	Babson College	US Central	Peter Cercone	11/07/2002	$15,600	$22,000	$6,600	%40
114144258	Ameribag Inc	US East	Leonard Miret	10/25/2002	$16,500	$21,000	$4,500	%80
113412623	PayChex	US West	Brett Henry	10/10/2002	$40,000	$44,000	$4,000	%10
214596989	AMCO	Retail	Samuel Stinson	09/22/2002	$14,000	$15,000	41,000	%15

Download: Excel Summary View | Excel Detail View | PDF Summary View

Appendix A.3 Summary View : VP Sales – US East

Summary View: VP Sales US East

Region: All Regions | Period: MOM

Districts: All Districts | Measure Type: Revenue

Territory: All Territories | Account Type: 9-Digit

Service Category: Over All

Decliners | Graphs

Acct #	Acct Name	Organization	Sales Professional	Date Last Called	Express Revenue ($)		Relative Change ($)	% Change
					Revenue Dec. 02	Rev. Nov. 02		
100078392	Sunoco Inc	Mid-Atlantic	Michael Clarke	10/11/2002	$24,000	$35,000	$11,000	%20
108571969	Energy Argus	Mid-Atlantic	Steve Smith	10/21/2002	$800	$9,800	$9,000	%30
133098429	Stanley Creations	Penn-Ohio	Pamela Lampkin	09/30/2002	$34,000	$42,500	$8,500	%50
258894766	Balchem Corp	Eastern Region	Carol Collins	11/17/2002	$21,000	$29,000	$8,000	%10
104229921	Balmar Inc	Mid-Atlantic	Lori Till	11/02/2002	$5,600	$13,000	$7,400	%80
180639691	Square One Comm	Penn-Ohio	Richard Solimeo	10/08/2002	$11,000	$18,300	$7,300	%29
104297340	Babson College	North-Eastern Re	Peter Cercone	11/07/2002	$15,600	$22,000	$6,600	%55
114144258	Ameribag Inc	Eastern Region	Leonard Miret	10/25/2002	$16,500	$21,000	$4,500	%48
113412623	PayChex	Penn-Ohio	Brett Henry	10/10/2002	$40,000	$44,000	$4,000	%11
214596989	AMCO	Penn-Ohio	Samuel Stinson	09/22/2002	$14,000	$15,000	41,000	%2

Download: Excel Summary View | Excel Detail View | PDF Summary View

Appendix A.4 Summary View : RSD – The River Valley

Summary View: RSD The River Valley

Districts: All Districts | Period: MOM

Territory: All Territories | Measure Type: Revenue

Account Type: 9-Digit

Service Category: Over All

	Decliners	Graphs						
Acct #	Acct Name	Organization	Sales Professional	Date Last Called	Express Revenue ($)		Relative Change ($)	% Change
					Revenue Dec. 02	Rev. Nov. 02		
100078392	Sunoco Inc	Gateway	Michael Clarke	10/11/2002	$24,000	$35,000	$11,000	%20
108571969	Energy Argus	Pioneer	Steve Smith	10/21/2002	$800	$9,800	$9,000	%30
133098429	Stanley Creations	Kansas	Pamela Lampkin	09/30/2002	$34,000	$42,500	$8,500	%50
258894766	Balchem Corp	Missouri	Carol Collins	11/17/2002	$21,000	$29,000	$8,000	%10
104229921	Balmar Inc	Bluff City	Lori Till	11/02/2002	$5,600	$13,000	$7,400	%80
180639691	Square One Comm	Tennessee Valley	Richard Solimeo	10/08/2002	$11,000	$18,300	$7,300	%29
104297340	Babson College	Mid-South	Peter Cercone	11/07/2002	$15,600	$22,000	$6,600	%55
114144258	Ameribag Inc	Pioneer	Leonard Miret	10/25/2002	$16,500	$21,000	$4,500	%48
113412623	PayChex	Gateway	Brett Henry	10/10/2002	$40,000	$44,000	$4,000	%11
214596989	AMCO	Bluff City	Samuel Stinson	09/22/2002	$14,000	$15,000	41,000	%2

Download: Excel Summary View | Excel Detail View | PDF Summary View

Appendix A.5 Summary View : DSM – Kansas District

Summary View: DSM Kansas District

Territory: All Territories | Period: MOM | Measure Type: Revenue | Account Type: 9-Digit | Service Category: Over All

Decliners | Graphs

Acct #	Acct Name	Organization	Sales Professional	Date Last Called	Express Revenue ($)		Relative Change ($)	% Change
					Revenue Dec. 02	Rev. Nov. 02		
100078392	Sunoco Inc	11-11-11-11-20	Michael Clarke	10/11/2002	$24,000	$35,000	$11,000	%40
108571969	Energy Argus	11-11-11-11-42	Steve Smith	10/21/2002	$800	$9,800	$9,000	%10
133098429	Stanley Creations	11-11-11-11-40	Pamela Lampkin	09/30/2002	$34,000	$42,500	$8,500	%80
258894766	Balchem Corp	11-11-11-11-25	Carol Collins	11/17/2002	$21,000	$29,000	$8,000	%29
104229921	Balmar Inc	11-11-11-11-20	Lori Till	11/02/2002	$5,600	$13,000	$7,400	%73
180639691	Square One Comm	11-11-11-11-40	Richard Solimeo	10/08/2002	$11,000	$18,300	$7,300	%25
104297340	Babson College	11-11-11-11-42	Peter Cercone	11/07/2002	$15,600	$22,000	$6,600	%36
114144258	Amenbag Inc	11-11-11-11-25	Leonard Miret	10/25/2002	$16,500	$21,000	$4,500	%38
113412623	PayChex	11-11-11-11-42	Brett Henry	10/10/2002	$40,000	$44,000	$4,000	%60
214596989	AMCO	11-11-11-11-20	Samuel Stinson	09/22/2002	$14,000	$15,000	41,000	%50

Download: Excel Summary View | Excel Detail View | PDF Summary View

Appendix A.6 Summary View : AE

Summary View: AE

Period: MOM

Measure Type: Revenue

Account Type: 9-Digit

Service Category: Over All

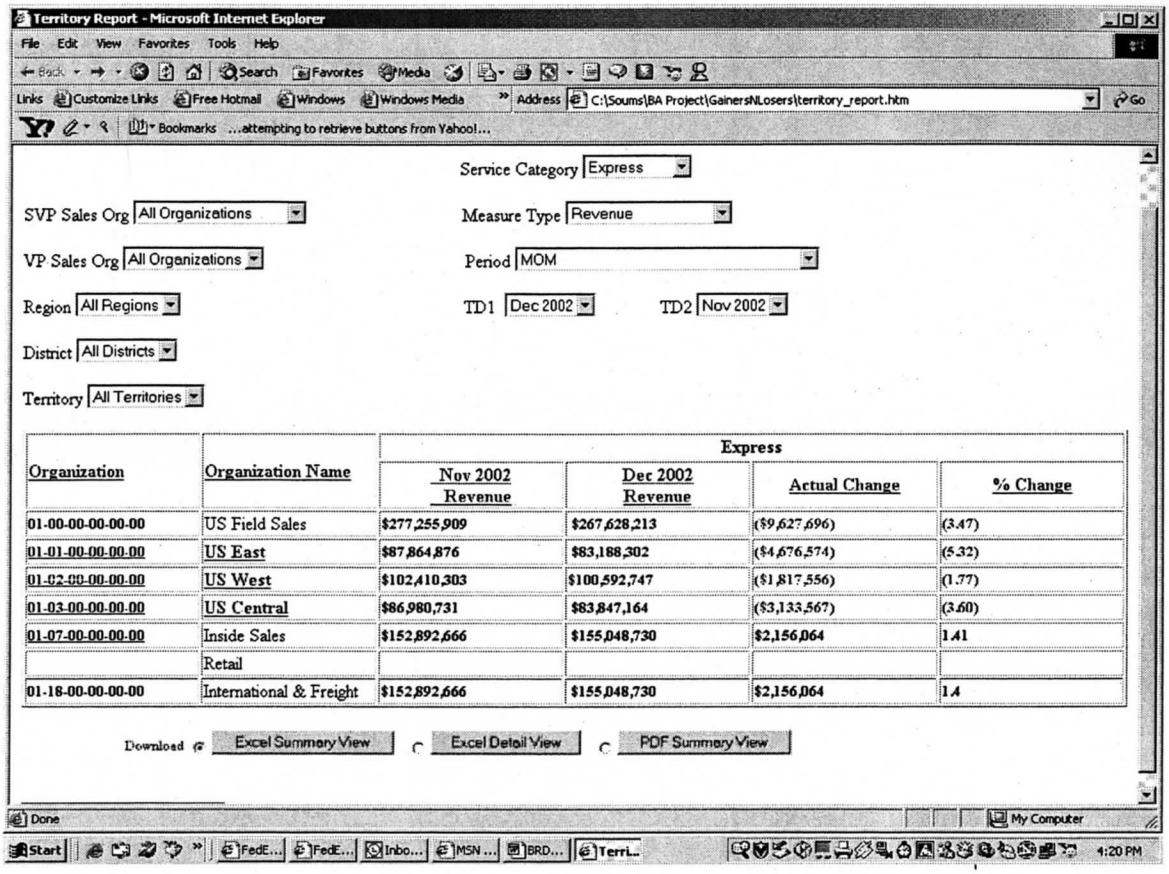
Decliners | Graphs

Acct #	Acct Name	Date Last Called	Express Revenue ($)		Relative Change ($)	% Change
			Revenue Dec. 02	Rev. Nov. 02		
100078392	Sunoco Inc	10/11/2002	$24,000	$35,000	$11,000	%36
108571969	Energy Argus	10/21/2002	$800	$9,800	$9,000	%64
133098429	Stanley Creations	09/30/2002	$34,000	$42,500	$8,500	%36
258894766	Balchem Corp	11/17/2002	$21,000	$29,000	$8,000	%29
104229921	Balmar Inc	11/02/2002	$5,600	$13,000	$7,400	%80
180639691	Square One Comm	10/08/2002	$11,000	$18,300	$7,300	%55
104297340	Babson College	11/07/2002	$15,600	$22,000	$6,600	%40
114144258	Amerbag Inc	10/25/2002	$16,500	$21,000	$4,500	%93
113412623	PayChex	10/10/2002	$40,000	$44,000	$4,000	%11
214596989	AMCO	09/22/2002	$14,000	$15,000	41,000	%15

Download: Excel Summary View | Excel Detail View | PDF Summary View

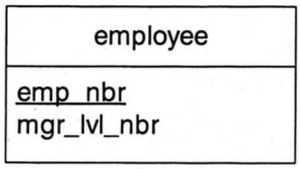
Appendix A.7: Territory Report View

Period Chosen: MOM
Measure Type: Revenue
Service Category: Express



Organization	Organization Name	Express			
		Nov 2002 Revenue	Dec 2002 Revenue	Actual Change	% Change
01-00-00-00-00-00	US Field Sales	$277,255,909	$267,628,213	($9,627,696)	(3.47)
01-01-00-00-00-00	US East	$87,864,876	$83,188,302	($4,676,574)	(5.32)
01-02-00-00-00-00	US West	$102,410,303	$100,592,747	($1,817,556)	(1.77)
01-03-00-00-00-00	US Central	$86,980,731	$83,847,164	($3,133,567)	(3.60)
01-07-00-00-00-00	Inside Sales	$152,892,666	$155,048,730	$2,156,064	1.41
	Retail				
01-18-00-00-00-00	International & Freight	$152,892,666	$155,048,730	$2,156,064	1.4

Appendix A.8: Employee Table



Appendix A.9: Customer Employee Assignment Table

customer_employee_assignment
sales_div_nbr
sales_grp_nbr
sales_org_nbr
sales_rgn_nbr
sales_dist_nbr
sales_terr_nbr
cust_nbr
cust_nm
cust_acct_crte_dt
date_last_called
emp_nbr
emp_nm
job_cd
natl_acct_nbr
natl_acct_nm
shp_pay_ind

Appendix A.10: Sales Territory Table

sales_territory
sales_div_nbr
sales_grp_nbr
sales_org_nbr
sales_rgn_nbr
sales_dist_nbr
sales_terr_nbr
sales_div_nm
sales_grp_nm
sales_org_nm
sales_rgn_nm
sales_dist_nm

Appendix A.11: Growth & Ranking Weekly Customer Revenue Summary Table

grow_rank_week_cust_rev_sum
year_nbr
mth_nbr
wk_end_dt
sales_div_nbr
sales_grp_nbr
sales_org_nbr
sales_rgn_nbr
sales_dist_nbr
sales_terr_nbr
cust_nbr
svc_cat_cd
svc_bas_cd
natl_acct_nbr
date_last_called
cust_acct_crte_dt
natl_acct_nm
cust_nm
emp_nbr
emp_nm
shp_pce_qty
adj_net_rev_amt
create_dt_tmst

Appendix A.12: Growth & Ranking Quarterly Customer Revenue Summary Table

grow_rank_qtr_cust_rev_sum
year_nbr
qtr_nbr
sales_div_nbr
sales_grp_nbr
sales_org_nbr
sales_rgn_nbr
sales_dist_nbr
sales_terr_nbr
cust_nbr
svc_cat_cd
svc_bas_cd
natl_acct_nbr
date_last_called
cust_acct_crte_dt
natl_acct_nm
cust_nm
emp_nbr
emp_nm
shp_pce_qty
adj_net_rev_amt
create_dt_tmst

Appendix A.13: Growth & Ranking Weekly Territory Revenue Summary Table

grow_rank_week_terr_rev_sum
year_nbr
mth_nbr
wk_end_dt
sales_div_nbr
sales_grp_nbr
sales_org_nbr
sales_rgn_nbr
sales_dist_nbr
sales_terr_nbr
svc_cat_cd
svc_bas_cd
shp_pce_qty
adj_net_rev_amt
create_dt_tmst

Appendix A.14: Growth & Ranking Quarterly Territory Revenue Summary Table

grow_rank_qtr_terr_rev_sum
year_nbr
qtr_nbr
sales_div_nbr
sales_grp_nbr
cust_nbr
sales_org_nbr
sales_rgn_nbr
sales_dist_nbr
sales_terr_nbr
svc_cat_cd
svc_bas_cd
shp_pce_qty
adj_net_rev_amt
create_dt_tmst

Appendix B – SQL Code

STEP 1

Extract data from each source table rolling up revenue, and piece quantity at the payr_cust_nbr, shipper_cust_nbr, sales month-yyyymm, trans_border_cd, svc_bas_cd, originating zip/country, and destination zip/country.

Domestic Shipment Initial Extract:

```
CREATE SET TABLE PROTO_DB.summary_dom_ship_tier1 ,NO FALLBACK ,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    payr_cust_nbr INTEGER FORMAT '-z(9)',
    shp_cust_nbr INTEGER FORMAT '-z(9)',
    shp_rate_yyyymm CHAR(6) CHARACTER SET LATIN NOT CASESPECIFIC,
    svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    pkg_typ_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    cnsgn_flg CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    both_flg CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    orig_pstl_cd CHAR(9) CHARACTER SET LATIN NOT CASESPECIFIC COMPRESS 'US
',
    dest_pstl_cd CHAR(9) CHARACTER SET LATIN NOT CASESPECIFIC COMPRESS 'US
',
```

```
    shp_pce_qty DECIMAL(9,0) FORMAT '--z(16)9',
    shp_rate_wgt DECIMAL(11,2) FORMAT '--z(14)9.99',
    shp_wgt DECIMAL(11,2) FORMAT '--z(14)9.99',
    net_rev_amt DECIMAL(11,2) FORMAT '--z(14)9.99',
    fuel_surcharge_amt DECIMAL(11,2) FORMAT '--z(14)9.99')
PRIMARY INDEX ( payr_cust_nbr ,shp_cust_nbr ,shp_rate_yyyymm ,
svc_bas_cd ,pkg_typ_cd ,cnsgn_flg );
```

```
SELECT
 dom.payr_cust_nbr,
 shp_cust_nbr,
 CAST (CAST (file_extr_dt_tmst as DATE FORMAT 'YYYYMM') as CHAR(6)),
 dom.svc_bas_cd,
 dom.pkg_typ_cd,

 case
  when dom.recp_cust_nbr > 0 and
      dom.recp_cust_nbr = dom.payr_cust_nbr and
      dom.recp_cust_nbr <> dom.shp_cust_nbr
    then
     'Y'
  else
    'N'
end csng_flag,

case
  when dom.payr_cust_nbr = dom.shp_cust_nbr
    then
     'B'
  else
```

```
    'N'
  end both_payer_and_shipper,


  orig_pstl_cd,
  dest_pstl_cd,
  sum(case when adj_typ_cd in ('1','98') then shp_pce_qty else 0 end),
  sum(case when adj_typ_cd in ('1','98') then shp_wgt else 0 end),
  sum(net_rev_amt),
  sum(zeroifnull(spechand.shp_sphnd_chrg_amt))
from
  ui_ish_prod_db.dom_shipment dom
    left outer join
        ui_ish_prod_db.shp_x_spec_hand spechand
on
  dom.shp_nbr = spechand.shp_nbr and
  sphnd_cd = '310'
where
  (dom.adj_typ_cd in ('1','2','98') or adj_typ_cd is null) and
  (bil_typ_cd <> '2' or bil_typ_cd is null) and
  (dom.inln_cd <> 'NR' or dom.inln_cd is NULL) and
  (rev_doc_typ_cd <> '12' or rev_doc_typ_cd is NULL) and
group by 1,2,3,4,5,6,7,8,9;
```

International Shipment Extract

```
CREATE MULTISET TABLE ZENITH_PROTO_DB.summary_intl_ship_tier1 ,NO
FALLBACK ,
    NO BEFORE JOURNAL,
    NO AFTER JOURNAL
    (
     payr_cust_nbr INTEGER FORMAT '-z(9)',
     shp_cust_nbr INTEGER FORMAT '-z(9)',
     shp_rate_yyyymm CHAR(6) CHARACTER SET LATIN NOT CASESPECIFIC,
     tran_bdr_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     pkg_typ_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     cnsgn_flg CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
     both_flg CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
     orig_ctry_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     dest_ctry_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     shp_pce_qty DECIMAL(9,0) FORMAT '--z(16)9',
     shp_wgt DECIMAL(11,2) FORMAT '--z(14)9.99',
     net_rev_amt DECIMAL(11,2) FORMAT '--z(14)9.99',
     fuel_surcharge_amt DECIMAL(11,2) FORMAT '--z(14)9.99')
UNIQUE PRIMARY INDEX ( payr_cust_nbr ,shp_cust_nbr ,shp_rate_yyyymm ,
```

```
tran_bdr_cd ,svc_bas_cd ,pkg_typ_cd ,cnsgn_flg ,both_flg ,orig_ctry_cd ,
dest_ctry_cd );

SELECT
 intl.payr_cust_nbr,
 shp_cust_nbr,
 CAST (CAST (file_extr_dt_tmst as DATE FORMAT 'YYYYMM') as CHAR(6)),
 tran_bdr_cd,
 intl.svc_bas_cd,
 intl.pkg_typ_cd,
 case
  when intl.recp_cust_nbr > 0 and intl.recp_cust_nbr = intl.payr_cust_nbr and intl.recp_cust_nbr
<> intl.shp_cust_nbr
    then
     'Y'
    else
     'N'
 end,

 case
  when intl.payr_cust_nbr = intl.shp_cust_nbr
    then
     'B'
```

```
   else
    'N'
end,

orig_ctry_cd,
dest_ctry_cd,
sum(
   Case
    When adj_typ_cd In ('1','98')
     Then
      (
       Case
        When intl.svc_bas_cd In ('13', '14')
         Then
          bag_qty
        Else
          shp_pce_qty
       end
      )
     Else
      0
   end
  ),
```

```
 sum(
    Case When shp_wgt_um_cd = 'k'
     Then
      shp_wgt * 2.2046
     Else
      shp_wgt
    end
   ),
 sum(net_rev_amt),
 sum(zeroifnull(spechand.shp_sphnd_chrg_amt))

From
 ui_ish_prod_db.intl_shipment intl
 Left Outer Join
 ui_ish_prod_db.shp_x_spec_hand spechand
  On
   intl.shp_nbr = spechand.shp_nbr and
   sphnd_cd = '310'
Where
 (intl.adj_typ_cd In ('1','2','98') Or adj_typ_cd is null) And
 (bil_typ_cd <> '2' Or bil_typ_cd is null) And
 (intl.inln_cd <> 'NR' Or intl.inln_cd is NULL)
Group By 1,2,3,4,5,6,7,8,9,10;
```

Ground Shipment Extract

```
CREATE MULTISET TABLE ZENITH_PROTO_DB.summary_fxgnd_ship_tier1 ,NO
FALLBACK ,
    NO BEFORE JOURNAL,
    NO AFTER JOURNAL
    (
     payr_cust_nbr INTEGER FORMAT '-z(9)',
     shp_cust_nbr INTEGER FORMAT '-z(9)',
     shp_rate_yyyymm CHAR(6) CHARACTER SET LATIN NOT CASESPECIFIC,
     tran_bdr_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     pkg_typ_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     cnsgn_flg CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
     both_flg CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
     shp_pce_qty DECIMAL(9,0) FORMAT '--z(16)9',
     shp_wgt DECIMAL(11,2) FORMAT '--z(14)9.99',
     net_rev_amt DECIMAL(11,2) FORMAT '--z(14)9.99',
     fuel_surcharge_amt DECIMAL(11,2) FORMAT '--z(14)9.99')
UNIQUE PRIMARY INDEX ( payr_cust_nbr ,shp_cust_nbr ,shp_rate_yyyymm ,
tran_bdr_cd ,svc_bas_cd ,pkg_typ_cd ,cnsgn_flg ,both_flg );

SELECT
```

```
case when n_rec_typ in ('P', 'I', 'Q')
 then
  n_cust_nbr
 else
  0
end payr_cust_nbr,
case when n_rec_typ in ('P', 'O', 'T')
 then
  n_cust_nbr
 else
  0
end shp_cust_nbr,
case
  when substr(n_snap_shot_per,5,2) in ('01','02','03','04','05','06','07')
   then
    cast(cast(substr(n_snap_shot_per,1,6) as integer) - 95 as char(6))
   else
    cast(cast(substr(n_snap_shot_per,1,6) as integer) - 7 as char(6))
end,
case
 when n_prod_typ in ('400','401','402','403','420','421','430','431','432',
                     '433','435','450','451','452','406','407','408','409',
                     '425','453','491','492','691','700','720','410','415',
```

```
                '468','710','494','411','416','483','711','000','999')
  then
   '1'
 when n_prod_typ in ('493','465', '470', '471', '474', '475', '440')
  then
   '2'
 when n_prod_typ in ('480','790','785','469','478','479','480','484','500',
                '501','422','423','460','461','462','463','464','472',
                '473','476','477','481','495','496','424','480','482',
                '490','485')
  then
   '3'
 else
  'a'
end,
case
 when n_prod_typ in ('411','416','483','711')
  then
   '20'
 when n_prod_typ in ('410','415','468','710','494','469','478','479',
              '480', '484','500','501')
  or
    (n_prod_typ = '480' and n_ctry_cd = '124')
```

```
  then
    '3'
  when n_prod_typ in ('400','401','402','403','420','421','430','431',
                      '432','433','435','450','451','452','406','407',
                      '408','409','425','453','491','492','691','493',
                      '465','470','471','474','475','440','422','423',
                      '460','461','462','463','464','472','473','476',
                      '477','481','495','496','424','480','482','490',
                      '485')
    or
    (n_prod_typ = '480' and n_ctry_cd = '840')
  then
   '92'
  when n_prod_typ in ('700','720','790','785')
   then
    '90'
  when n_prod_typ = '000'
   then
    '98'
  when n_prod_typ = '999'
   then
    '97'
end,
```

```
'1' pkg_typ_cd,

case when n_rec_typ = 'I'
 then
  'Y'
 else
  'N'
end cnsgn_flg,

case when payr_cust_nbr = shp_cust_nbr
 then
  'B'
 else
  'N'
end both_flg,

sum(n_inv_wk_pkg_cnt),
sum(n_inv_wk_wgt),
sum(n_inv_wk_chrg_amt - n_inv_wk_dsc_amt + n_acs_wk_chrg_amt),
sum(n_acs_wk_fuel_amt)

FROM
```

```
(
SELECT
 case when  ti_ctry_cd is null
  then
   ta_ctry_cd
  else
   ti_ctry_cd
 end ctry_cd ,

 case when ti_cust_nbr is null
  then
   ta_cust_nbr
  else
   ti_cust_nbr
 end cust_nbr,

 case when ti_snap_shot_per is null
  then
   ta_snap_shot_per
  else
   ti_snap_shot_per
 end snap_shot_per,
```

```
case when ti_prod_typ  is null
 then
   ta_prod_typ
 else
   ti_prod_typ
end,

case when ti_rec_typ  is null
 then
   ta_rec_typ
 else
   ti_rec_typ
end rec_typ,
zeroifnull(ti_inv_wk_pkg_cnt),
zeroifnull(ti_inv_wk_chrg_amt),
zeroifnull(ti_inv_wk_dsc_amt),
zeroifnull(ti_inv_wk_wgt),
zeroifnull(ta_acs_wk_chrg_amt),
zeroifnull(ta_acs_wk_fuel_amt)

FROM
```

```
(
 SELECT
  inv.ctry_cd,
  inv.cust_nbr,
  inv.snap_shot_per,
  inv.prod_typ,
  inv.rec_typ,
  sum(inv.wk_pkg_cnt),
  sum(inv.wk_chrg_amt),
  sum(inv.wk_dsc_amt),
  sum(inv.wk_wgt)
 FROM
  ui_ish_prod_db.rps_invoice inv
 WHERE
  (
   inv.wk_pkg_cnt > 0 or
   inv.wk_chrg_amt > 0 or
   inv.wk_dsc_amt > 0 or
   inv.wk_wgt > 0
  ) and
  inv.snap_shot_per in
   (
    SELECT
```

```
    snap_shot_per
   FROM
    ui_ish_prod_db.rps_bus_cal
   WHERE
    substr(snap_shot_per,1,6)
     between '200201' and '200312'
  )
 GROUP by 1,2,3,4,5
)
 temp_invoice
 (
  ti_ctry_cd,
  ti_cust_nbr,
  ti_snap_shot_per,
  ti_prod_typ,
  ti_rec_typ,
  ti_inv_wk_pkg_cnt,
  ti_inv_wk_chrg_amt,
  ti_inv_wk_dsc_amt,
  ti_inv_wk_wgt
 )

FULL OUTER JOIN
```

```
(
 SELECT
  acs.ctry_cd,
  acs.cust_nbr,
  acs.snap_shot_per,
  acs.prod_typ,
  acs.rec_typ,
  sum(
     case when acs_cd not in ('240','241')
       then
         acs.wk_chrg_amt
       else
         0
     end
   ),
  sum(
     case when acs_cd = '325'
       then
         acs.wk_chrg_amt
       else
         0
     end
```

```
) fuel_surcharge_amt
FROM
 ui_ish_prod_db.rps_inv_acs acs
WHERE
 (
  acs.wk_pkg_cnt > 0 or
  acs.wk_chrg_amt > 0
 ) and
 snap_shot_per in

 (
   SELECT
    snap_shot_per
   FROM
    ui_ish_prod_db.rps_bus_cal
   WHERE
    substr(snap_shot_per,1,6)
      between '200201' and '200312'
GROUP by 1,2,3,4,5
)
temp_acs
(
 ta_ctry_cd,
```

```
    ta_cust_nbr,
    ta_snap_shot_per,
    ta_prod_typ,
    ta_rec_typ,
    ta_acs_wk_chrg_amt,
    ta_acs_wk_fuel_amt
   )
  ON
   ta_ctry_cd = ti_ctry_cd and
   ta_cust_nbr = ti_cust_nbr and
   ta_snap_shot_per = ti_snap_shot_per and
   ta_prod_typ = ti_prod_typ and
   ta_rec_typ = ti_rec_typ
)
 HONKING
(
 n_ctry_cd,
 n_cust_nbr,
 n_snap_shot_per,
 n_prod_typ,
 n_rec_typ,
 n_inv_wk_pkg_cnt,
 n_inv_wk_chrg_amt,
```

```
  n_inv_wk_dsc_amt,
  n_inv_wk_wgt,
  n_acs_wk_chrg_amt,
  n_acs_wk_fuel_amt
 )
GROUP by 1,2,3,4,5,6,7,8;
```

STEP 1A

Sum data out of Step 1 inserting into TIER1 tables

Domestic Shipment Tier1 Build (three steps insert):

```
CREATE MULTISET TABLE ZENITH_PROTO_DB.dom_exp_cust_rev_summary ,NO
FALLBACK ,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    year_nbr SMALLINT,
    mth_nbr SMALLINT,
    cust_nbr INTEGER,
    shp_pay_ind CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    pkg_typ_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    cnsgn_flg CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    orig_pstl_cd VARCHAR(9) CHARACTER SET LATIN NOT CASESPECIFIC,
    dest_pstl_cd VARCHAR(9) CHARACTER SET LATIN NOT CASESPECIFIC,
    shp_wgt DECIMAL(11,2),
    shp_pce_qty DECIMAL(9,0),
    net_rev_amt DECIMAL(11,2),
```

```
    fuel_srchg_amt DECIMAL(11,2),
    create_dt_tmst TIMESTAMP(0) NOT NULL)
UNIQUE PRIMARY INDEX ( year_nbr ,mth_nbr ,cust_nbr ,shp_pay_ind ,
svc_bas_cd ,pkg_typ_cd ,cnsgn_flg ,orig_pstl_cd ,dest_pstl_cd );

insert into zenith_proto_db.dom_exp_cust_rev_summary
sel
    substr(a.shp_rate_yyyymm,1,4),
    substr(a.shp_rate_yyyymm,5,2),
    a.payr_cust_nbr,
    a.both_flg,

    a.svc_bas_cd,
    a.pkg_typ_cd,
    a.cnsgn_flg,

    a.orig_pstl_cd,
    a.dest_pstl_cd,
    sum(shp_wgt),
    sum(shp_pce_qty),
    sum(net_rev_amt),
    sum(fuel_surcharge_amt),
    cast ('${timeStamp}' as timestamp(0))
```

```
 from
   proto_db.summary_dom_ship_tier1 a
  where both_flg = 'B'
  group by 1,2,3,4,5,6,7,8,9;

insert into zenith_proto_db.dom_exp_cust_rev_summary
  sel
    substr(a.shp_rate_yyyymm,1,4),
    substr(a.shp_rate_yyyymm,5,2),
    a.payr_cust_nbr,
    'P',

    a.svc_bas_cd,
    a.pkg_typ_cd,
    a.cnsgn_flg,

    a.orig_pstl_cd,
    a.dest_pstl_cd,
    sum(shp_wgt),
    sum(shp_pce_qty),
    sum(net_rev_amt),
    sum(fuel_surcharge_amt) ,
    cast ('${timeStamp}' as timestamp(0))
```

```
from
 proto_db.summary_dom_ship_tier1 a
where both_flg = 'N'
group by 1,2,3,4,5,6,7,8,9;

insert into zenith_proto_db.dom_exp_cust_rev_summary
SELECT
  substr(a.shp_rate_yyyymm,1,4),
  substr(a.shp_rate_yyyymm,5,2),
  a.shp_cust_nbr,
  'S',

  a.svc_bas_cd,
  a.pkg_typ_cd,
  a.cnsgn_flg,

  a.orig_pstl_cd,
  a.dest_pstl_cd,
  sum(shp_wgt),
  sum(shp_pce_qty),
  sum(net_rev_amt),
  sum(fuel_surcharge_amt) ,
```

```
  cast ('${timeStamp}' as timestamp(0))


from

 proto_db.summary_dom_ship_tier1 a

 where both_flg = 'N'

group by 1,2,3,4,5,6,7,8,9;
```

International Shipment Tier1 Build (three steps insert):

```
CREATE SET TABLE ZENITH_PROTO_DB.intl_exp_cust_rev_summary ,NO FALLBACK ,
    NO BEFORE JOURNAL,
    NO AFTER JOURNAL
    (
     year_nbr SMALLINT,
     mth_nbr SMALLINT,
     cust_nbr INTEGER,
     shp_pay_ind CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
     svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     pkg_typ_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     tran_bdr_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
     cnsgn_flg CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     orig_ctry_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     dest_ctry_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     shp_wgt DECIMAL(11,2),
     shp_pce_qty DECIMAL(9,0),
     net_rev_amt DECIMAL(11,2),
     fuel_srchg_amt DECIMAL(11,2),
     create_dt_tmst TIMESTAMP(0) NOT NULL)
UNIQUE PRIMARY INDEX ( year_nbr ,mth_nbr ,cust_nbr ,shp_pay_ind ,
```

```
svc_bas_cd ,pkg_typ_cd ,tran_bdr_cd ,cnsgn_flg ,orig_ctry_cd ,
dest_ctry_cd );


insert into ZENITH_PROTO_DB.intl_exp_cust_rev_summary
sel
    substr(a.shp_rate_yyyymm,1,4),
    substr(a.shp_rate_yyyymm,5,2),
    a.payr_cust_nbr,
    a.both_flg,

    a.svc_bas_cd,
    a.pkg_typ_cd,
    a.tran_bdr_cd,
    a.cnsgn_flg,

    a.orig_ctry_cd,
    a.dest_ctry_cd,
    sum(shp_wgt),
    sum(shp_pce_qty),
    sum(net_rev_amt),
    sum(fuel_surcharge_amt),
    cast ('${timeStamp}' as timestamp(0))
```

```
 from
   ZENITH_PROTO_DB.summary_intl_ship_tier1 a
  where both_flg = 'B'
  group by 1,2,3,4,5,6,7,8,9,10;

insert into ZENITH_PROTO_DB.intl_exp_cust_rev_summary
  sel
    substr(a.shp_rate_yyyymm,1,4),
    substr(a.shp_rate_yyyymm,5,2),
    a.payr_cust_nbr,
    'P',

    a.svc_bas_cd,
    a.pkg_typ_cd,
    a.tran_bdr_cd,
    a.cnsgn_flg,

    a.orig_ctry_cd,
    a.dest_ctry_cd,
    sum(shp_wgt),
    sum(shp_pce_qty),
    sum(net_rev_amt),
    sum(fuel_surcharge_amt) ,
```

```
    cast ('${timeStamp}' as timestamp(0))


  from
    ZENITH_PROTO_DB.summary_intl_ship_tier1 a
  where both_flg = 'N'
  group by 1,2,3,4,5,6,7,8,9,10;
insert into ZENITH_PROTO_DB.intl_exp_cust_rev_summary
SELECT
    substr(a.shp_rate_yyyymm,1,4),
    substr(a.shp_rate_yyyymm,5,2),
    a.shp_cust_nbr,
    'S',

    a.svc_bas_cd,
    a.pkg_typ_cd,
    a.tran_bdr_cd,
    a.cnsgn_flg,

    a.orig_ctry_cd,
    a.dest_ctry_cd,
    sum(shp_wgt),
    sum(shp_pce_qty),
    sum(net_rev_amt),
```

```
  sum(fuel_surcharge_amt) ,
  cast ('${timeStamp}' as timestamp(0))

from
  ZENITH_PROTO_DB.summary_intl_ship_tier1 a
where both_flg = 'N'
group by 1,2,3,4,5,6,7,8,9,10;
```

Ground Shipment Tier1 Build (three step insert):

```
CREATE SET TABLE ZENITH_PROTO_DB.fxgnd_cust_rev_summary ,NO FALLBACK ,
  NO BEFORE JOURNAL,
  NO AFTER JOURNAL
  (
   year_nbr SMALLINT,
   mth_nbr SMALLINT,
   cust_nbr INTEGER,
   shp_pay_ind CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
   svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
   pkg_typ_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
   tran_bdr_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
   cnsgn_flg CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
   shp_wgt DECIMAL(11,2),
```

```
    shp_pce_qty DECIMAL(9,0),
    net_rev_amt DECIMAL(11,2),
    fuel_srchg_amt DECIMAL(11,2),
    create_dt_tmst TIMESTAMP(0) NOT NULL)
UNIQUE PRIMARY INDEX ( year_nbr ,mth_nbr ,cust_nbr ,shp_pay_ind ,
svc_bas_cd ,pkg_typ_cd ,tran_bdr_cd ,cnsgn_flg );

insert into zenith_proto_db.fxgnd_cust_rev_summary

sel
    substr(a.shp_rate_yyyymm,1,4),
    substr(a.shp_rate_yyyymm,5,2),
    a.payr_cust_nbr,
    a.both_flg,

    a.svc_bas_cd,
    a.pkg_typ_cd,
    a.tran_bdr_cd,
    a.cnsgn_flg,

    sum(shp_wgt),
    sum(shp_pce_qty),
    sum(net_rev_amt),
```

```
  sum(fuel_surcharge_amt),
  cast ('${timeStamp}' as timestamp(0))
from
  zenith_proto_db.summary_fxgnd_ship_tier1 a
where both_flg = 'B' and payr_cust_nbr > 0
group by 1,2,3,4,5,6,7,8;

insert into zenith_proto_db.fxgnd_cust_rev_summary

sel
  substr(a.shp_rate_yyyymm,1,4),
  substr(a.shp_rate_yyyymm,5,2),
  a.payr_cust_nbr,
  'P',

  a.svc_bas_cd,
  a.pkg_typ_cd,
  a.tran_bdr_cd,
  a.cnsgn_flg,

  sum(shp_wgt),
  sum(shp_pce_qty),
  sum(net_rev_amt),
```

```
  sum(fuel_surcharge_amt) ,
  cast ('${timeStamp}' as timestamp(0))

 from
  zenith_proto_db.summary_fxgnd_ship_tier1 a
 where both_flg = 'N' and payr_cust_nbr > 0
 group by 1,2,3,4,5,6,7,8;

insert into zenith_proto_db.fxgnd_cust_rev_summary
SELECT
  substr(a.shp_rate_yyyymm,1,4),
  substr(a.shp_rate_yyyymm,5,2),
  a.shp_cust_nbr,
  'S',

  a.svc_bas_cd,
  a.pkg_typ_cd,
  a.tran_bdr_cd,
  a.cnsgn_flg,

  sum(shp_wgt),
  sum(shp_pce_qty),
  sum(net_rev_amt),
```

```
  sum(fuel_surcharge_amt) ,
  cast ('${timeStamp}' as timestamp(0))

from
  zenith_proto_db.summary_fxgnd_ship_tier1 a
 where both_flg = 'N' and shp_cust_nbr > 0
group by 1,2,3,4,5,6,7,8;
```

Ground 9 Digit Shipment Tier1 Build:

```
CREATE SET TABLE ZENITH_PROTO_DB.fxgnd_9_dig_cust_rev_summary ,NO
FALLBACK ,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    year_nbr SMALLINT,
    mth_nbr SMALLINT,
    cust_nbr INTEGER,
    shp_pay_ind CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    pkg_typ_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    tran_bdr_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
```

```
    cnsgn_flg CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    shp_wgt DECIMAL(11,2),
    shp_pce_qty DECIMAL(9,0),
    net_rev_amt DECIMAL(11,2),
    fuel_srchg_amt DECIMAL(11,2),
    create_dt_tmst TIMESTAMP(0) NOT NULL)
UNIQUE PRIMARY INDEX ( year_nbr ,mth_nbr ,cust_nbr ,shp_pay_ind ,
svc_bas_cd ,pkg_typ_cd ,tran_bdr_cd ,cnsgn_flg );

del from zenith_proto_db.fxgnd_9_dig_cust_rev_summary;

insert into zenith_proto_db.fxgnd_9_dig_cust_rev_summary
SELECT
    year_nbr,
    mth_nbr,
    b.parent_cust_nbr,
    a.shp_pay_ind,

    a.svc_bas_cd,
    a.pkg_typ_cd,
    a.tran_bdr_cd,
    a.cnsgn_flg,
```

```
  shp_wgt,
  shp_pce_qty,
  net_rev_amt,
  fuel_srchg_amt ,
  create_dt_tmst

from
  zenith_proto_db.fxgnd_cust_rev_summary a,
  ui_ish_prod_db.fedex_rps_xref b
where
  b.child_cust_nbr = a.cust_nbr and
  b.child_acct_type = 'RP';
```

International Next Flight Extract

```
CREATE SET TABLE ZENITH_PROTO_DB.intl_next_flt_cust_rev_summary ,NO
FALLBACK ,
  NO BEFORE JOURNAL,
  NO AFTER JOURNAL
  (
   year_nbr SMALLINT,
   mth_nbr SMALLINT,
   cust_nbr INTEGER,
```

```
    shp_pay_ind CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    pkg_typ_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    cnsgn_flg CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    orig_ctry_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    dest_ctry_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    shp_wgt DECIMAL(11,2),
    shp_pce_qty DECIMAL(9,0),
    net_rev_amt DECIMAL(11,2),
    fuel_srchg_amt DECIMAL(11,2),
    create_dt_tmst TIMESTAMP(0) NOT NULL)
UNIQUE PRIMARY INDEX ( year_nbr ,mth_nbr ,cust_nbr ,shp_pay_ind ,
svc_bas_cd ,pkg_typ_cd ,orig_ctry_cd ,dest_ctry_cd );


SELECT
 substr(cast(cast(inf.inv_dt as date format 'yyyymm') as char(6)),1,4),
 substr(cast(cast(inf.inv_dt as date format 'yyyymm') as char(6)),5,2),
 inf.payr_cust_nbr,
 case
  when cust_role_cd = 'S'
   then
    'B'
```

```
    else
      'P'
  end,
  svc_base_cd,
  pkg_type_cd,
  case
    when cust_role_cd = 'C'
    then
      'Y'
    else
      'N'
  end,
  orig_cntry_cd,
  dest_cntry_cd,
  sum(shp_wgt),
  sum(shp_piece_qty),
  sum(net_rev_amt),
  sum(case when spechand.shp_srchg_amt is not null then spechand.shp_srchg_amt else 0 end),
  cast ('${timeStamp}' as timestamp(0))
from
  ui_ish_prod_db.intl_next_flt_shp inf
 left outer join
  ui_ish_prod_db.intl_next_flt_x_srchg     spechand
```

```
 on
  inf.shp_nbr = spechand.shp_nbr and
  srchg_cd = '856'
where
 (cast(inf.inv_dt As date format 'yyyymm') between
   add_months( cast (current_date As date format 'yyyymm'), -24) and
   add_months (cast (current_date As date format 'yyyymm'), -1))
group by 1,2,3,4,5,6,7,8,9;
```

Same Day Shipment Extract

```
CREATE SET TABLE ZENITH_PROTO_DB.same_day_cust_rev_summary ,NO FALLBACK
,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    year_nbr SMALLINT,
    mth_nbr SMALLINT,
    cust_nbr INTEGER,
    shp_pay_ind CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    pkg_typ_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    cnsgn_flg CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    orig_pstl_cd VARCHAR(9) CHARACTER SET LATIN NOT CASESPECIFIC,
    dest_pstl_cd VARCHAR(9) CHARACTER SET LATIN NOT CASESPECIFIC,
    shp_wgt DECIMAL(11,2),
    shp_pce_qty DECIMAL(9,0),
    net_rev_amt DECIMAL(11,2),
    fuel_srchg_amt DECIMAL(11,2),
    create_dt_tmst TIMESTAMP(0) NOT NULL)
UNIQUE PRIMARY INDEX ( year_nbr ,mth_nbr ,cust_nbr ,shp_pay_ind ,
```

```
svc_bas_cd ,pkg_typ_cd ,cnsgn_flg ,orig_pstl_cd ,dest_pstl_cd );


SELECT
 substr(cast(cast(SD.inv_dt as date format 'yyyymm') as char(6)),1,4),
 substr(cast(cast(SD.inv_dt as date format 'yyyymm') as char(6)),5,2),
 SD.payr_cust_nbr,
 case
  when cust_rol_cd = 'S'
   then
    'B'
   else
    'P'
 end,
 svc_typ_cd,
 '1',
 case
  when cust_rol_cd = 'C'
   then
    'Y'
   else
    'N'
 end,
 pckup_pstl_cd,
```

```
  recp_pstl_cd,
  sum(shp_wgt),
  sum(shp_pce_qty),
  sum(net_rev_amt),
  sum(case when spechand.shp_srchg_amt is not null then spechand.shp_srchg_amt else 0 end),
  cast ('${timeStamp}' as timestamp(0))
from
  ui_ish_prod_db.sameday_shp SD
 left outer join
  ui_ish_prod_db.sameday_shp_x_srchg spechand
on
  SD.shp_nbr = spechand.shp_nbr and
  srchg_cd = '856'
where
  cast(sd.inv_dt As date format 'yyyymm') between
    add_months( cast (current_date As date format 'yyyymm'), -24) and
    add_months (cast (current_date As date format 'yyyymm'), -1)
group by 1,2,3,4,5,6,7,8,9;
```

STEP 2

Summarize all sources into one table (5 step insert process)

```
CREATE MULTISET TABLE ZENITH_PROTO_DB.cust_rev_summary_tier2 ,NO
FALLBACK ,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    source_of_data CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    year_nbr SMALLINT,
    mth_nbr SMALLINT,
    cust_nbr INTEGER,
    shp_pay_ind CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    svc_cat_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    shp_wgt DECIMAL(11,2),
    shp_pce_qty DECIMAL(9,0),
    net_rev_amt DECIMAL(11,2),
    fuel_srchg_amt DECIMAL(11,2))
PRIMARY INDEX ( year_nbr ,mth_nbr ,cust_nbr ,shp_pay_ind ,svc_cat_cd ,
svc_bas_cd );
```

```
del from  ZENITH_PROTO_DB.cust_rev_summary_tier2;


insert into ZENITH_PROTO_DB.cust_rev_summary_tier2
sel
'dom',
year_nbr,
mth_nbr,
cust_nbr,
shp_pay_ind,
b.svc_category,
a.svc_bas_cd,
sum(shp_wgt),
sum(shp_pce_qty),
sum(net_rev_amt),
sum(fuel_srchg_amt) from
ZENITH_PROTO_DB.dom_exp_cust_rev_summary a
left outer join ZENITH_PROTO_DB.svc_category_xref b
on a.svc_bas_cd = b.svc_bas_cd and b.tran_bdr_cd = 1
group by 1,2,3,4,5,6,7;


insert into ZENITH_PROTO_DB.cust_rev_summary_tier2
```

```
sel
'int',
year_nbr,
mth_nbr,
cust_nbr,
shp_pay_ind,
b.svc_category,
a.svc_bas_cd,
sum(shp_wgt),
sum(shp_pce_qty),
sum(net_rev_amt),
sum(fuel_srchg_amt) from
ZENITH_PROTO_DB.intl_exp_cust_rev_summary a
left outer join ZENITH_PROTO_DB.svc_category_xref b
on a.svc_bas_cd = b.svc_bas_cd and a.tran_bdr_cd = b.tran_bdr_cd
group by 1,2,3,4,5,6,7;

insert into ZENITH_PROTO_DB.cust_rev_summary_tier2
sel
'gnd',
t1.year_nbr,
t1.mth_nbr,
t1.parent_cust_nbr,
```

```
t1.shp_pay_ind,
b.svc_category,
t1.svc_bas_cd,
sum(shp_wgt),
sum(shp_pce_qty),
sum(net_rev_amt),
sum(fuel_srchg_amt) from
(
sel
year_nbr,
mth_nbr,
temp1.parent_cust_nbr,
shp_pay_ind,
a.svc_bas_cd,
a.tran_bdr_cd,
sum(shp_wgt),
sum(shp_pce_qty),
sum(net_rev_amt),
sum(fuel_srchg_amt) from
ZENITH_PROTO_DB.fxgnd_cust_rev_summary a,
(sel child_cust_nbr, min(parent_cust_nbr) from ui_ish_prod_Db.fedex_rps_xref w
'RP' group by 1) temp1 (child_cust_nbr, parent_cust_nbr)
where a.cust_nbr = temp1.child_cust_nbr
```

```
group by 1,2,3,4,5,6) t1 (year_nbr,
mth_nbr,
parent_cust_nbr,
shp_pay_ind,
svc_bas_cd,
tran_bdr_cd,
shp_wgt,
shp_pce_qty,
net_rev_amt,
fuel_srchg_amt)
left outer join ZENITH_PROTO_DB.svc_category_xref b
on t1.svc_bas_cd = b.svc_bas_cd and t1.tran_bdr_cd = b.tran_bdr_cd
group by 1,2,3,4,5,6,7;

insert into ZENITH_PROTO_DB.cust_rev_summary_tier2
sel
'INF',
year_nbr,
mth_nbr,
cust_nbr,
shp_pay_ind,
b.svc_category,
a.svc_bas_cd,
```

```
sum(shp_wgt),
sum(shp_pce_qty),
sum(net_rev_amt),
sum(fuel_srchg_amt) from
ZENITH_PROTO_DB.intl_next_flt_cust_rev_summary a
left outer join ZENITH_PROTO_DB.svc_category_xref b
on a.svc_bas_cd = b.svc_bas_cd and b.tran_bdr_cd = '2'
group by 1,2,3,4,5,6,7;


insert into ZENITH_PROTO_DB.cust_rev_summary_tier2
sel
'SD ',
year_nbr,
mth_nbr,
cust_nbr,
shp_pay_ind,
b.svc_category,
a.svc_bas_cd,
sum(shp_wgt),
sum(shp_pce_qty),
sum(net_rev_amt),
sum(fuel_srchg_amt) from
```

ZENITH_PROTO_DB.same_day_cust_rev_summary a

left outer join ZENITH_PROTO_DB.svc_category_xref b

on a.svc_bas_cd = b.svc_bas_cd and b.tran_bdr_cd = '1'

group by 1,2,3,4,5,6,7;

Note: Where tran_bdr is not available in tier1 table the join condition is hard coded

STEP 2A

Summarize cust_rev_summary_tier2 table by year_nbr, mth_nbr, cust_nbr, shp_pay_ind, svc_cat_cd, svc_bas_cd

```
CREATE SET TABLE ZENITH_PROTO_DB.cust_rev_summary ,NO FALLBACK ,
    NO BEFORE JOURNAL,
    NO AFTER JOURNAL
    (
     year_nbr SMALLINT,
     mth_nbr SMALLINT,
     cust_nbr INTEGER,
     shp_pay_ind CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
     svc_cat_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
     svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
     shp_pce_qty DECIMAL(9,0),
     net_rev_amt DECIMAL(11,2),
     fuel_srchg_amt DECIMAL(11,2),
     create_dt_tmst TIMESTAMP(0) NOT NULL)
PRIMARY INDEX ( year_nbr ,mth_nbr ,cust_nbr ,shp_pay_ind ,svc_cat_cd ,
svc_bas_cd );
```

```
del from  ZENITH_PROTO_DB.cust_rev_summary;


insert into ZENITH_PROTO_DB.cust_rev_summary
sel
year_nbr,
mth_nbr,
cust_nbr,
shp_pay_ind,
svc_cat_cd,
svc_bas_cd,
sum(shp_pce_qty),
sum(net_rev_amt),
sum(fuel_srchg_amt) ,
cast ('${timeStamp}' as timestamp(0))
from
ZENITH_PROTO_DB.cust_rev_summary_tier2
group by 1,2,3,4,5,6;
```

STEP 3A

Here we have build a temporary table to pull in all data elements needed to build the Actuate Growth and Ranking tables as well as Tier3 table.

```
CREATE SET TABLE ZENITH_PROTO_DB.customer_employee_assignment ,NO
FALLBACK ,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    sales_div_nbr INTEGER,
    sales_grp_nbr INTEGER,
    SALEs_ORG_NBR INTEGER,
    sales_RGN_NBR INTEGER,
    sales_AREA_NBR INTEGER,
    sales_DIST_NBR INTEGER,
    sales_TERR_NBR INTEGER,
    cust_nbr INTEGER,
    CUST_NM VARCHAR(30) CHARACTER SET LATIN NOT CASESPECIFIC,
    cust_acct_Crte_dt DATE FORMAT 'YYYY-MM-DD',
    DATE_LAST_CALLED DATE FORMAT 'YYYY-MM-DD',
    EMP_NBR VARCHAR(10) CHARACTER SET LATIN NOT CASESPECIFIC,
```

```
    EMP_NM VARCHAR(50) CHARACTER SET LATIN NOT CASESPECIFIC,
    JOB_CD CHAR(5) CHARACTER SET LATIN NOT CASESPECIFIC,
    NATL_ACCT_NBR INTEGER,
    NATL_ACCT_NM VARCHAR(30) CHARACTER SET LATIN NOT CASESPECIFIC,
    shp_pay_ind CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC)
PRIMARY INDEX dgoradt ( sales_div_nbr ,sales_grp_nbr ,SALEs_ORG_NBR ,
sales_RGN_NBR ,sales_AREA_NBR ,sales_DIST_NBR ,sales_TERR_NBR ,
CUST_NBR );

del from $ZENITH_DB.customer_employee_assignment;

INSERT INTO $ZENITH_DB.customer_employee_assignment
select
 ca.div_nbr,
 ca.grp_nbr,
 ca.sale_org_nbr,
 ca.rgn_nbr,
 ca.area_nbr,
 ca.dist_nbr,
 ca.terr_nbr,
 ca.cust_nbr,
 cust.cust_nm,
 cust.cust_acct_crte_dt,
```

```
 sci.date_last_called,
 ea.emp_nbr,
 trim (both from emp.first_nm) || ' ' || trim (both from emp.last_nm),
 emp.job_cd,
 natl.natl_acct_nbr,
 natl.natl_acct_nm,
 case
  when (ca.sale_org_nbr in (1,2,3,18) or
      (ca.sale_org_nbr = 7  and ca.rgn_nbr in (20,21,22)) or
      (ca.sale_org_nbr = 12 and ca.rgn_nbr = 31) or
      (ca.sale_org_nbr = 13 and ca.rgn_nbr = 34))
    then
     'S'
  else
    'P'
 end
from
 (
  sel
   sales_div_nbr,
   sales_grp_nbr,
   sales_org_nbr,
   sales_rgn_nbr,
```

```
  sales_area_nbr,
  sales_dist_nbr,
  sales_terr_nbr,
  cust_nbr
 from
   ui_ish_prod_db.business_alignment
 where
  current_date between eff_dt and exp_dt and
  sales_org_nbr in (1,2,3,7,12,13,14,15,16,17,18,20)
 group by 1,2,3,4,5,6,7,8
)
 ca
(
 div_nbr,
 grp_nbr,
 sale_org_nbr,
 rgn_nbr,
 area_nbr,
 dist_nbr,
 terr_nbr,
 cust_nbr
)
left outer join
```

```
 ui_ish_prod_db.employee_assignment ea
on
 ca.sale_org_nbr = ea.sale_org_nbr and
 ca.rgn_nbr = ea.rgn_nbr and
 ca.area_nbr = ea.area_nbr and
 ca.dist_nbr = ea.dist_nbr and
 ca.terr_nbr = ea.terr_nbr and
 current_date between ea.eff_dt and ea.exp_dt and
 ea.align_type_cd = 'a'

left outer join
(
 sel
  cust_nbr,
  emp_nbr,
  cast(max(cust_intr_tmst) as date)
 from
  ui_ish_prod_db.sales_call_interactions
 group by 1,2
)
 sci
(
 cust_nbr,
```

```
 emp_nbr,
 date_last_called
)
on
 sci.cust_nbr = ca.cust_nbr and
 sci.emp_nbr = ea.emp_nbr
left outer join
 ui_ish_prod_db.customer cust
on
 ca.cust_nbr = cust.cust_nbr
left outer join
 ui_ish_prod_db.employee emp
on
 ea.emp_nbr = emp.emp_nbr

left outer join
 ui_ish_prod_db.national_acct natl
on
 cust.natl_acct_nbr = natl.natl_acct_nbr;
```

STEP 3B

Tier3 Build:

```
CREATE MULTISET TABLE ZENITH_PROTO_DB.sales_org_mth_cust_rev_summary ,NO
FALLBACK ,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    year_nbr SMALLINT,
    mth_nbr SMALLINT,
    sales_div_nbr INTEGER,
    sales_grp_nbr INTEGER,
    sales_org_nbr INTEGER,
    sales_rgn_nbr INTEGER,
    sales_area_nbr INTEGER,
    sales_dist_nbr INTEGER,
    sales_terr_nbr INTEGER,
    cust_nbr INTEGER,
    svc_cat_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    svc_bas_cd CHAR(3) CHARACTER SET LATIN NOT CASESPECIFIC,
    shp_pce_qty DECIMAL(9,0),
    net_rev_amt DECIMAL(11,2),
```

```
    fuel_srchg_amt DECIMAL(11,2),
    create_dt_tmst TIMESTAMP(0))
PRIMARY INDEX ( year_nbr ,mth_nbr ,sales_div_nbr ,sales_grp_nbr ,
sales_org_nbr ,sales_rgn_nbr ,sales_area_nbr ,sales_dist_nbr ,
sales_terr_nbr ,cust_nbr ,svc_cat_cd ,svc_bas_cd );

delete from ZENITH_PROTO_DB.sales_org_mth_cust_rev_summary

insert into ZENITH_PROTO_DB.sales_org_mth_cust_rev_summary
sel
  crs.year_nbr,
  crs.mth_nbr,
  cea.sales_div_nbr,
  cea.sales_grp_nbr,
  cea.sales_org_nbr,
  cea.sales_rgn_nbr,
  cea.sales_area_nbr,
  cea.sales_dist_nbr,
  cea.sales_terr_nbr,
  cea.cust_nbr,
  crs.svc_cat_cd,
  crs.svc_bas_cd,
  sum(
```

```
    case
     when cea.shp_pay_ind = 'S' and
        crs.shp_pay_ind in ('S','B')
     then
       shp_pce_qty
     when cea.shp_pay_ind = 'P' and
        crs.shp_pay_ind in ('P','B')
     then
       shp_pce_qty
     else
      0
    end
  ) pce_qty,

sum (
    case
     when cea.shp_pay_ind = 'S' and
        crs.shp_pay_ind in ('S','B')
      then
       net_rev_amt
     when cea.shp_pay_ind = 'P' and
        crs.shp_pay_ind in ('P','B')
      then
```

```
        net_rev_amt
      else
        0
    end),
  sum(
    case
      when cea.shp_pay_ind = 'S' and
          crs.shp_pay_ind in ('S','B')
        then
          fuel_srchg_amt
      when cea.shp_pay_ind = 'P' and
          crs.shp_pay_ind in ('P','B')
        then
          fuel_srchg_amt
      else
        0
    end
  ),
  cast(current_timestamp(0) as timestamp(0))
from
 cust_rev_summary crs,
 customer_employee_assignment cea
```

```
where
   crs.cust_nbr = cea.cust_nbr
group by 1,2,3,4,5,6,7,8,9,10,11,12;
```

Growth and Ranking Specific Tables

```
CREATE SET TABLE ZENITH_PROTO_DB.GROW_RANK_MTH_CUST_REV_SUM ,NO
FALLBACK ,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    year_nbr SMALLINT,
    mth_nbr SMALLINT,
    sales_div_nbr INTEGER,
    sales_grp_nbr INTEGER,
    sales_org_nbr INTEGER,
    sales_rgn_nbr INTEGER,
    sales_area_nbr INTEGER,
    sales_dist_nbr INTEGER,
    sales_terr_nbr INTEGER,
    natl_acct_nbr INTEGER,
    cust_nbr INTEGER,
    date_last_called DATE FORMAT 'YYYY-MM-DD',
```

```
    cust_acct_crte_dt DATE FORMAT 'YYYY-MM-DD',
    svc_cat_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    natl_acct_nm VARCHAR(30) CHARACTER SET LATIN NOT CASESPECIFIC,
    cust_nm VARCHAR(30) CHARACTER SET LATIN NOT CASESPECIFIC,
    emp_nbr VARCHAR(10) CHARACTER SET LATIN NOT CASESPECIFIC,
    emp_nm VARCHAR(30) CHARACTER SET LATIN NOT CASESPECIFIC,
    shp_pce_qty DECIMAL(9,0),
    adj_net_rev_amt DECIMAL(11,2),
    create_dt_tmst TIMESTAMP(0))
PRIMARY INDEX ymdgoradtcs ( year_nbr ,mth_nbr ,sales_div_nbr ,
sales_grp_nbr ,sales_org_nbr ,sales_rgn_nbr ,sales_area_nbr ,
sales_dist_nbr ,sales_terr_nbr ,cust_nbr ,svc_cat_cd );

del from $ZENITH_DB.GROW_RANK_MTH_CUST_REV_SUM;
insert into  $ZENITH_DB.GROW_RANK_MTH_CUST_REV_SUM
SELECT
 crs.year_nbr,
 crs.mth_nbr,
 cea.sales_div_nbr,
 cea.sales_grp_nbr,
 cea.sales_org_nbr,
 cea.sales_rgn_nbr,
 cea.sales_area_nbr,
```

```
  cea.sales_dist_nbr,
  cea.sales_terr_nbr,
  cea.natl_acct_nbr,
  crs.cust_nbr,
  cea.date_last_called,
  cea.cust_acct_crte_dt,
  crs.svc_cat_cd,
  cea.natl_acct_nm,
  cea.cust_nm,
  cea.emp_nbr,
  cea.emp_nm,
  crs.shp_pce_qty,
  crs.net_rev_amt - crs.fuel_srchg_amt,
  cast(current_timestamp(0) as timestamp(0))
from
  $ZENITH_DB.cust_rev_summary crs,
  $ZENITH_DB.customer_employee_assignment cea
where
  cea.cust_nbr = crs.cust_nbr
group by 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20;
```

```
CREATE SET TABLE ZENITH_PROTO_DB.GROW_RANK_QTR_CUST_REV_SUM ,NO
FALLBACK ,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    year_nbr SMALLINT,
    fscl_qtr_nbr SMALLINT,
    sales_div_nbr INTEGER,
    sales_grp_nbr INTEGER,
    sales_org_nbr INTEGER,
    sales_rgn_nbr INTEGER,
    sales_area_nbr INTEGER,
    sales_dist_nbr INTEGER,
    sales_terr_nbr INTEGER,
    natl_acct_nbr INTEGER,
    cust_nbr INTEGER,
    date_last_called DATE FORMAT 'YYYY-MM-DD',
    cust_acct_crte_dt DATE FORMAT 'YYYY-MM-DD',
    svc_cat_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    natl_acct_nm VARCHAR(30) CHARACTER SET LATIN NOT CASESPECIFIC,
    cust_nm VARCHAR(30) CHARACTER SET LATIN NOT CASESPECIFIC,
    emp_nbr VARCHAR(10) CHARACTER SET LATIN NOT CASESPECIFIC,
    emp_nm VARCHAR(30) CHARACTER SET LATIN NOT CASESPECIFIC,
```

```
    shp_pce_qty DECIMAL(9,0),
    adj_net_rev_amt DECIMAL(11,2),
    create_dt_tmst TIMESTAMP(0))
PRIMARY INDEX ymdgoradtcs ( year_nbr ,fscl_qtr_nbr ,sales_div_nbr ,
sales_grp_nbr ,sales_org_nbr ,sales_rgn_nbr ,sales_area_nbr ,
sales_dist_nbr ,sales_terr_nbr ,cust_nbr ,svc_cat_cd );


del from $ZENITH_DB.GROW_RANK_QTR_CUST_REV_SUM;
insert into  $ZENITH_DB.GROW_RANK_QTR_CUST_REV_SUM
SELECT
 case
  when mth_nbr in (1,2,3,4,5)
   then
    year_nbr
   else
    year_nbr + 1
end,
case
  when mth_nbr in (6,7,8)
   then
    1
  when mth_nbr in (9,10,11)
```

```
  then
    2
  when mth_nbr in (12,1,2)
  then
    3
  when mth_nbr in (3,4,5)
  then
    4
end,
sales_div_nbr,
sales_grp_nbr,
sales_org_nbr,
sales_rgn_nbr,
sales_area_nbr,
sales_dist_nbr,
sales_terr_nbr,
natl_acct_nbr,
cust_nbr,
date_last_called,
cust_acct_crte_dt,
svc_cat_cd,
natl_acct_nm,
cust_nm,
```

```
  emp_nbr,
  emp_nm,
  sum(shp_pce_qty),
  sum(adj_net_rev_amt),
  cast(current_timestamp(0) as timestamp(0))
from
  $ZENITH_DB.GROW_RANK_MTH_CUST_REV_SUM
group by 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18;
```

```
CREATE SET TABLE ZENITH_PROTO_DB.GROW_RANK_MTH_TERR_REV_SUM ,NO
FALLBACK ,
   NO BEFORE JOURNAL,
   NO AFTER JOURNAL
   (
    year_nbr SMALLINT,
    mth_nbr SMALLINT,
    sales_div_nbr INTEGER,
    sales_grp_nbr INTEGER,
    sales_org_nbr INTEGER,
    sales_rgn_nbr INTEGER,
    sales_area_nbr INTEGER,
    sales_dist_nbr INTEGER,
    sales_terr_nbr INTEGER,
    svc_cat_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
    shp_pce_qty DECIMAL(9,0),
    adj_net_rev_amt DECIMAL(11,2),
    create_dt_tmst TIMESTAMP(0))
PRIMARY INDEX ymdgoradtcs ( year_nbr ,mth_nbr ,sales_div_nbr ,
sales_grp_nbr ,sales_org_nbr ,sales_rgn_nbr ,sales_area_nbr ,
sales_dist_nbr ,sales_terr_nbr ,svc_cat_cd );
```

```
del from $ZENITH_DB.GROW_RANK_MTH_TERR_REV_SUM;

insert into  $ZENITH_DB.GROW_RANK_MTH_TERR_REV_SUM

SELECT

 year_nbr,

 mth_nbr,

 sales_div_nbr,

 sales_grp_nbr,

 sales_org_nbr,

 sales_rgn_nbr,

 sales_area_nbr,

 sales_dist_nbr,

 sales_terr_nbr,

 svc_cat_cd,

 sum(shp_pce_qty),

 sum(adj_net_rev_amt),

 cast(current_timestamp(0) as timestamp(0))

from

 $ZENITH_DB.GROW_RANK_MTH_CUST_REV_SUM

group by 1,2,3,4,5,6,7,8,9,10;
```

```
CREATE SET TABLE ZENITH_PROTO_DB.GROW_RANK_QTR_TERR_REV_SUM ,NO
FALLBACK ,
    NO BEFORE JOURNAL,
    NO AFTER JOURNAL
    (
     year_nbr SMALLINT,
     fscl_qtr_nbr SMALLINT,
     sales_div_nbr INTEGER,
     sales_grp_nbr INTEGER,
     sales_org_nbr INTEGER,
     sales_rgn_nbr INTEGER,
     sales_area_nbr INTEGER,
     sales_dist_nbr INTEGER,
     sales_terr_nbr INTEGER,
     svc_cat_cd CHAR(1) CHARACTER SET LATIN NOT CASESPECIFIC,
     shp_pce_qty DECIMAL(9,0),
     adj_net_rev_amt DECIMAL(11,2),
     create_dt_tmst TIMESTAMP(0))
PRIMARY INDEX ymdgoradtcs ( year_nbr ,fscl_qtr_nbr ,sales_div_nbr ,
sales_grp_nbr ,sales_org_nbr ,sales_rgn_nbr ,sales_area_nbr ,
sales_dist_nbr ,sales_terr_nbr ,svc_cat_cd );
```

```
del from $ZENITH_DB.GROW_RANK_QTR_TERR_REV_SUM;

insert into  $ZENITH_DB.GROW_RANK_QTR_TERR_REV_SUM

SELECT
 case
  when mth_nbr in (1,2,3,4,5)
   then
    year_nbr
   else
    year_nbr + 1
 end,
 case
  when mth_nbr in (6,7,8)
   then
    1
  when mth_nbr in (9,10,11)
   then
    2
  when mth_nbr in (12,1,2)
   then
    3
  when mth_nbr in (3,4,5)
```

```
    then
      4
  end,
  sales_div_nbr,
  sales_grp_nbr,
  sales_org_nbr,
  sales_rgn_nbr,
  sales_area_nbr,
  sales_dist_nbr,
  sales_terr_nbr,
  svc_cat_cd,
  sum(shp_pce_qty),
  sum(adj_net_rev_amt),
  cast(current_timestamp(0) as timestamp(0))
from
  $ZENITH_DB.GROW_RANK_MTH_TERR_REV_SUM
group by 1,2,3,4,5,6,7,8,9,10;
```